

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Інститут телекомунікаційних систем
Кафедра Телекомунікаційних систем**

«На правах рукопису»
УДК 621.319

«До захисту допущено»

Завідувач кафедри

_____ Л.О. Уривський

« ____ » _____ 20__ р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 172 Телекомунікації та радіотехніка

**на тему: «Розробка автоматизованої системи реєстрації та обробки
показників обладнання обліку енергоносіїв»**

Виконав:

студент II курсу, групи ТС-81мп
Матюшенко Данило Максимович

Керівник:

Доцент кафедри ТС, к.т.н., доцент
Осипчук С.О.

Рецензент:

Доцент кафедри ТК, к.т.н., доцент
Міночкін Д.А.

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студент (-ка) _____

Київ – 2019 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Інститут телекомунікаційних систем

Кафедра Телекомунікаційних систем

Рівень вищої освіти – другий (магістерський) за освітньо-професійною програмою

Спеціальність (спеціалізація) – 172 «Телекомунікації та радіотехніка» (172.3620.1 «Телекомунікаційні системи та мережі»)

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Л.О. Уривський

« ____ » _____ 20__ р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Матюшенко Данилу Максимовичу

1. Тема дисертації **«Розробка автоматизованої системи реєстрації та обробки показників обладнання обліку енергоносіїв»**, науковий керівник дисертації Осипчук Сергій Олександрович, кандидат технічних наук, асистент кафедри Телекомунікаційних систем Інституту телекомунікаційних систем, затверджені наказом по університету від « ____ » _____ 20__ р. № _____

2. Термін подання студентом дисертації _____

3. Об'єкт дослідження: **технології реєстрації, обробки і передавання даних** в застосунках обліку енергоносіїв ЖКГ

4. Предмет дослідження: **прикладні аспекти реалізації рішень і способів реєстрації, обробки і передавання даних** в застосунках обліку енергоносіїв ЖКГ

5. Перелік завдань, які потрібно розробити: **розробка автоматизованої системи реєстрації, обробки і передавання в центр обробки даних відповідних показників обладнання обліку енергоносіїв ЖКГ, з метою**

економічно доцільної модернізації існуючого обладнання обліку енергоносіїв ЖКГ без заміни обладнання, та з автоматизацією процесу передавання даних

6. Орієнтовний перелік графічного (ілюстративного) матеріалу

Плакат № 1 «Тема, мета, актуальність, об'єкт, предмет, проблематика, завдання дослідження»»

Плакат № 2 «Порівняння вартості існуючої системи із власною розробкою»

Плакат № 3 «Технології передачі даних»

Плакат № 4 «Порівняльний аналіз плат для розробки системи»

Плакат № 5 «Архітектура»

Плакат № 6 «Ілюстрація установки»

Плакат № 7 «Отримані результати»

Плакат № 8 «Висновки»

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка

Студент

Матюшенко Д. М.

Науковий керівник дисертації

Осипчук С. О.

РЕФЕРАТ

Темою магістерської дисертації є розробка автоматизованої системи реєстрації та обробки показників обладнання обліку енергоносіїв.

Робота містить 86 сторінок, зокрема 23 ілюстрацій, 3 таблиці та 7 джерел інформації.

Тема магістерської роботи є актуальною, оскільки досі на території України немає бюджетних рішень у автоматизованих системах реєстрації та обробці показників обладнання обліку енергоносіїв. Доволі часто ще й доступ до обладнання обліку є ускладненим, тому ця проблема дуже актуальна.

Мета дисертації полягає в формуванні рекомендацій щодо вибору інформаційно-телекомунікаційних технологій реєстрації, обробки і передавання даних, із врахуванням прикладних аспектів реалізації рішень і способів реєстрації, обробки і передавання даних в застосунках обліку енергоносіїв ЖКГ.

Об'єктом дослідження є прикладні аспекти реалізації рішень і способів реєстрації, обробки і передавання даних в застосунках обліку енергоносіїв ЖКГ.

Прикладним завданням у магістерській роботі стала розробка автоматизованої системи реєстрації, обробки і передавання в центр обробки даних відповідних показників обладнання обліку енергоносіїв ЖКГ, з метою економічно доцільної модернізації існуючого обладнання обліку енергоносіїв ЖКГ без заміни обладнання, та з автоматизацією процесу передавання даних.

В ході роботи було розглянуто декілька технологій передавання інформації: в тому числі GSM, Ethernet, LoRa та Wi-Fi. Також багато

разів змінювалось обладнання і залізо для роботи: починалось все з плати ESP8266 та невдовзі вона перестала задовільняти потреби, було також декілька невдалих спроб перенести всі напрацювання на плати, геть іншого типу, які використовують мову програмування MicroPython, але і ця ідея ідея потерпіла фіаско через недостатню розповсюдженість цих плат на Українських просторах та комплектуючих до них. Після цього було вирішено повернутись до мови плат типу Arduino: була взята плата ESP32-CAM від виробника AiThinker, яка повністю задовольняє усім потребам і є по суті готовим «кітом» для наших цілей.

У роботі була використана середа програмування мікроконтролерів «ArduinoIDE». Для передачі даних рекомендується в роботі використовувати Wi-Fi з'єднання до мережі інтернет. Були надані рекомендації щодо вибору обладнання, топології мережі, протоколів маршрутизації.

ABSTRACT

The topic of the master's thesis is the development of an automated system for registration and processing of indicators of energy metering equipment.

The work contains 86 pages, including 23 illustrations, 3 tables and 7 sources of information.

The topic of the master's thesis is relevant, as there are still no budget decisions in the territory of Ukraine in the automated systems of registration and processing of indicators of energy metering equipment. Quite often, access to accounting equipment is also difficult, so this problem is very urgent.

The purpose of the dissertation is to formulate recommendations on the choice of information and telecommunication technologies for registration, processing and transmission of data, taking into account the applied aspects of the implementation of decisions and methods of registration, processing and transmission of data in the applications of energy accounting of utilities.

The object of the study is the applied aspects of the implementation of solutions and methods of registration, processing and transfer of data in the applications of energy metering utilities.

An applied task in the master's work was the development of an automated system for registration, processing and transfer to the data processing center of relevant indicators of equipment for the accounting of energy utilities, with the aim of economically feasible modernization of existing energy metering equipment without replacement of equipment, and with automation of the data transmission process.

Several information transfer technologies were considered during the work: including GSM, Ethernet, LoRa and Wi-Fi. The hardware and iron for the job were also changed many times: everything started with the ESP8266

board and soon it stopped satisfying the needs, there were also several unsuccessful attempts to transfer all the proceeds to the boards, apart from another type, using the MicroPython programming language, but this idea also failed. due to the lack of distribution of these boards in the Ukrainian territories and their accessories. After that, it was decided to go back to the language of Arduino cards: the ESP32-CAM card from the manufacturer AiThinker was taken, which completely satisfies all needs and is essentially a ready-made "cat" for our purposes.

The programming environment used for programming the ArduinoIDE microcontrollers. For data transfer it is recommended to use Wi-Fi internet connection in work. Recommendations were given on equipment selection, network topology, routing protocols.

РЕФЕРАТ

Темой магистерской диссертации является разработка автоматизированной системы регистрации и обработки показателей оборудования учета энергоносителей.

Работа содержит 86 страниц, в том числе 23 иллюстраций, 3 таблицы и 7 источников информации.

Тема магистерской диссертации является актуальной, поскольку до сих пор на территории Украины нет бюджетных решений в автоматизированных системах регистрации и обработке показателей оборудования учета энергоносителей. Довольно часто еще и доступ к оборудованию учета затруднен, поэтому эта проблема очень актуальна.

Цель работы заключается в формировании рекомендаций по выбору информационно-телекоммуникационных технологий регистрации, обработки и передачи данных, с учетом прикладных аспектов реализации решений и способов регистрации, обработки и передачи данных в приложениях учета энергоносителей ЖКХ.

Объектом исследования являются прикладные аспекты реализации решений и способов регистрации, обработки и передачи данных в приложениях учета энергоносителей ЖКХ.

Прикладными задачами в магистерской работе стала разработка автоматизированной системы регистрации, обработки и передачи в центр обработки данных соответствующих показателей оборудования учета энергоносителей ЖКХ, с целью экономически целесообразной модернизации существующего оборудования учета энергоносителей ЖКХ без замены оборудования, и с автоматизацией процесса передачи данных.

В ходе работы были рассмотрены несколько технологий передачи информации: в том числе GSM, Ethernet, LoRa и Wi-Fi. Также много раз менялось оборудование и железо для работы: начиналось все с платы ESP8266 и вскоре она перестала удовлетворять потребности, было также несколько неудачных попыток перенести все наработки на платы, совершенно другого типа, которые используют язык программирования MicroPython, но и эта идея потерпела фиаско из-за недостаточной распространенности этих плат на украинском пространстве и комплектующих к ним. После этого было решено вернуться к языку плат типа Arduino: взята плата ESP32-CAM от производителя AiThinker, которая полностью удовлетворяет всем потребностям и является по сути готовым «китом» для наших целей.

В работе была использована среда программирования микроконтроллеров «ArduinoIDE». Для передачи данных рекомендуется в работе использовать Wi-Fi соединение к сети интернет. Были даны рекомендации по выбору оборудования, топологии сети, протоколов маршрутизации.

ПЕРЕЛІК СКОРОЧЕНЬ	11
ВСТУП	14
РОЗДІЛ 1. АНАЛІЗ ВПЛИВУ СХЕМИ ПОБУДОВИ МЕРЕЖІ НА ЇЇ НАДІЙНІСТЬ.....	16
1.1 Аналіз літературних джерел	16
1.2 Пропозиції на ринку.....	24
1.3 Приблизна вартість проектів.....	30
1.4 Висновки з розділу 1	32
РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ ОБЛІКУ ЕНЕРГОНОСІЇВ	33
2.1 Вибір технології для передачі даних	33
2.2 Вибір обладнання для проекту	38
2.3 Вибір плати	41
2.3 Архітектура проекту	56
2.4 Висновки з розділу 2.....	59
РОЗДІЛ 3. МАТЕРІАЛЬНА БАЗА	61
3.1 Raspberry	61
РОЗДІЛ 4. РОБОТА СИСТЕМИ.....	69
4.1 Програмування ESP32-CAM.....	69
4.2 Налаштування Raspberry	76
ВИСНОВКИ.....	85
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	86

ПЕРЕЛІК СКОРОЧЕНЬ

ЖКГ – житлово комунальне господарство

GSM - Global System for Mobile Communications

LoRa - Long Range

Wi-Fi - Wireless Fidelity

ЛУЗОД - Локальне устаткування збору і обробки даних

АСКОЕ – автоматизована система комерційного обліку електричної енергії

ПКЕЕ _ПРАВИЛА користування електричною енергією

SIM - Subscriber Identification Module

ГП – державне підприємство

СМС - Short Message Service

ПК – персональний комп'ютер

АРМ - *Автоматизированное рабочее место*

WEB (англ. *web* — паутина)

АСТОЕ - Система автоматизованого технічного обліку електроенергії

ПЗ – програмне забезпечення

WPAN - Wireless personal area network

WLAN - Wireless Local Area Network

WMAN - Wireless Metropolitan Area Networks

WWAN - *Wireless Wide Area Network*

EDGE - Enhanced Data rates for GSM Evolution

HSPA - *High Speed Packet Access*

IEEE - *Institute of Electrical and Electronics Engineers*

3G - third generation

LED - Light-emitting diode

IDE - Integrated Development Environment

USB - Universal Serial Bus

ICSP - In Circuit Serial Programming

AC/DC - alternating current/direct current

UART - Universal Asynchronous Receiver-Transmitter

FTDI - Future Technology Devices International

SRAM - static random access memory

EEPROM - Electrically Erasable Programmable Read-Only Memory

SPI - Serial Peripheral Interface

SS - Slave Select

MOSI - Master Out Slave In

MISO - Master In Slave Out

SCK - Serial Clock

AREF - Analogue REference

TWI - Two Wire Interface

SDA - data line

SCL - clock line

I2C - Inter-Integrated Circuit

PSRAM - Psuedo SRAM

SD - Secure Digital

FPC - Free Pascal Compiler

GFLOPS - FLoating-point Operations

HDTV - High Definition Television

GPIO - general-purpose input/output

OpenELEC - Open Embedded Linux Entertainment Center

OSMC - open source media center

HACA - National Aeronautics and Space Administration

http - HyperText Transfer Protocol

HTML - HyperText Markup Language

PHP - Hypertext Preprocessor

OCR - optical character recognition

ВСТУП

Лінь – це, мабуть, найбільший і найпрогресивніший двигун прогресу. Так, наприклад, для того аби не ходити і не перемикати канали на телевізорі «людина розумна» придумала пульт дистанційного керування. Так само були розроблені і рішення в сфері інтернет-речей: розумний будинок, системи автоматичного керування мікрокліматом, світолом, тощо. Сьогодні, вже, навіть, важко уявити повсякденне життя без сучасних засобів передачі інформації, гої вже й згадувати сфери науки, промислової діяльності, тощо.

Так само розвивалась і сфера житловом-комунального господарства: поява водопроводу, електрики, газу, інтрнету в кожну оселю. З'явилась величезна кількість обладнання та споживачів цих послуг. Але для контролю кількості спожитих послуг почали слідкувати не так давно. Нажаль, на сьогоднішній день далеко не всі оселі, будинки, квартири, підприємства доцільно використовують енергоносії. Далеко не всі можуть кожного дня по декілька разів заглядати на той чи інший лічильник з метою контролю своїх витрат.

Так от задля покращення і оптимізації використання енергоносіїв та ресурсів впроваджують автоматизовані систему обліку. Нажаль, на нашому ринку представлено буквально декілька готових рішень. Але їхня вартість є доволі великою, як для звичайного споживача.

Наша розробка дозволить, навіть, звичайному рядовому українцю моніторити свої витрати та заощаджувати ресурси. За допомогою цього, на перший, погляд простого приладу можна буде повністю автоматизувати збір інформації з усіх лічильників та проводити аналіз і не тільки.

По факту ця розробка є частиною, в майбутньому, кожного розумного будинку. Завдяки рішенням інтрнет-речей люди зможуть автоматизувати не тільки керування оселею, але і зможуть автоматизувати оплату за полуги житлово-комунального господарства але і мінімізувати витрати та зекономити свій час.

РОЗДІЛ 1. АНАЛІЗ ВПЛИВУ СХЕМИ ПОБУДОВИ МЕРЕЖІ НА ЇЇ НАДІЙНІСТЬ

1.1 Аналіз літературних джерел

“ЛУЗОД (локальне устаткування збору і обробки даних) – улаштована з метою проведення розрахунків за спожиту електричну енергію сукупність засобів обліку, які забезпечують вимірювання, збір, накопичення, оброблення інформації про обсяги і параметри потоків електричної енергії та значення споживаної потужності на окремій площадці вимірювання за відповідними періодами часу та мають інтерфейс дистанційного зчитування даних для роботи у складі автоматизованої системи комерційного обліку.

АСКОЕ (автоматизована система комерційного обліку електричної енергії) – сукупність об’єднаних в єдину систему локального устаткування збору і обробки даних (ЛУЗОД) засобів обліку, каналів передачі інформації та пристроїв приймання, обробки, відображення та реєстрації інформації.

Принципова відмінність ЛУЗОД від АСКОВЕ полягає в тому, що клієнт, у якого встановлено ЛУЗОД має можливість дистанційно переглядати дані щодо споживання електричної енергії лише у Особистому кабінеті КИЇВЕНЕРГО. Тоді як система АСКОВЕ дає можливість дистанційно переглядати та контролювати дані щодо споживання електричної енергії та потужності, користуючись програмним забезпеченням, яке встановлюється під час монтажу та налаштування АСКОВЕ.

Які переваги отримує клієнт від використання АСКОВЕ:

Можливість перегляду даних щодо споживання електричної енергії та потужності в будь-який момент та з будь-якою періодичністю.

Можливість здійснення аналізу споживання, прогнозування затрат на електроенергію.

Можливість зберігання та надання інформації щодо споживання електричної енергії та потужності в зручному для аналізу вигляді.

Можливість самостійно контролювати ліміти споживання електричної потужності в години максимального навантаження енергосистеми.

Можливість самостійно контролювати ліміти споживання електричної енергії.

Які споживачі зобов'язані облаштувати свої електроустановки ЛУЗОД або АСКОВЕ:

Відповідно до п. 3.35 ПОВЕ:

Діючі об'єкти, (крім багатоквартирних житлових будинків та населених пунктів) з приєднаною потужністю 150 кВт і більше та середньомісячним обсягом споживання за попередні 12 розрахункових періодів 50 тис. кВтг і більше.

Нові об'єкти з приєднаною потужністю 150 кВт і більше та середньомісячним заявленим обсягом споживання електричної енергії 50 тис. кВтг і більше.

Що необхідно зробити для того, щоб встановити на своєму об'єкті систему ЛУЗОД або АСКОВЕ

Відповідно до п. 3.8. ПОВЕ клієнт подає заяву до центру обслуговування клієнтів КИЇВЕНЕРГО з проханням надати технічні рекомендації щодо облаштування на своєму об'єкті ЛУЗОД/АСКОВЕ.

Бланк заяви на надання технічних рекомендацій щодо облаштування ЛУЗОД/АСКОВЕ.

Зразок заповнення заяви на надання технічних рекомендацій щодо облаштування ЛУЗОД/АСКОВЕ.

Технічні рекомендації містять:

перелік рекомендованих приладів обліку;

перелік місць установлення засобів обліку;

інформацію про параметри каналів зв'язку, які будуть застосовуватись для зчитування даних з ЛУЗОД або обміну даними з АСКОЕ клієнта.

Після отримання від КИЇВЕНЕРГО технічних рекомендацій, з їх урахуванням клієнтом розробляється технічне завдання для побудови ЛУЗОД або АСКОЕ.

Клієнт (проектна організація) надає розроблене технічне завдання до центру обслуговування клієнтів КИЇВЕНЕРГО . КИЇВЕНЕРГО протягом 10 робочих днів з дня отримання розглядає та погоджує його. У разі необхідності доопрацювання технічного завдання КИЇВЕНЕРГО надає свої зауваження та рекомендації з посиланнями на нормативні документи.

На основі погодженого технічного завдання клієнт (проектна організація) розробляє проект на встановлення ЛУЗОД/АСКОЕ (п. 3.12 ПКЕЕ).

Клієнт (проектна організація) надає розроблений проект до центру обслуговування клієнтів КИЇВЕНЕРГО. КИЇВЕНЕРГО протягом 15 робочих днів з дня отримання розглядає та погоджує його. У разі необхідності доопрацювання проекту КИЇВЕНЕРГО надає свої зауваження та рекомендації з посиланнями на нормативні документи.

Придбання обладнання та матеріалів (лічильники, обладнання передачі даних, клемні колодки, спеціалізовані шафи, кабельна продукція), а також проведення робіт з влаштування ЛУЗОД/АСКОЕ здійснюється клієнтом самостійно та за власні кошти. Роботи з влаштування ЛУЗОД/АСКОЕ виконуються підрядною організацією.

Перед встановленням лічильників електроенергії, вони мають пройти процедуру вхідного контролю та параметризації в КИЇВЕНЕРГО. Для цього клієнт звертається з заявою до центру обслуговування клієнтів КИЇВЕНЕРГО для отримання направлення на вхідний контроль та параметризацію. Послуга параметризації платна. В центрі обслуговування клієнт отримує рахунок на оплату послуги. Після пред'явлення до центру обслуговування сплаченої квитанції, йому видається направлення на вхідний контроль та параметризацію лічильника.

Бланк заяви на проходження вхідного контролю та парметризації.

Зразок заповнення заяви на проходження вхідного контролю та парметризації.

Роботи по вхідному контролю та параметризації лічильників проводяться за адресою: Залізничне шосе, 13.

З собою потрібно мати:

лічильники;

направлення на виконання робіт;

паспорти лічильників.

Для забезпечення доступу до первинної бази даних лічильників з боку КИЇВЕНЕРГО, клієнт має отримати SIM-картку. Для отримання SIM-картки клієнту необхідно звернутися до КИЇВЕНЕРГО за адресою: вул. Залізничне шосе, 13 та надати:

Заяву на отримання SIM-картки.

Зразок заповнення заяви на отримання SIM-картки.

Ксерокопію першої сторінки проекту на облаштування ЛУЗОД/АСКОЕ, погодженого з КИЇВЕНЕРГО;

Однолінійну схему живлення об'єкта клієнта;

Специфікацію приєднань об'єктів з зазначенням каналів зв'язку кожного лічильника.

Бланк специфікації приєднань об'єктів з зазначенням каналів зв'язку.

Після завершення робіт по влаштуванню ЛУЗОД/АСКОЕ клієнт (проектна або монтажна організація) складає акт прийняття в дослідну експлуатацію.

Для введення обладнання в дослідну експлуатацію клієнту потрібно надати до центру обслуговування клієнтів КИЇВЕНЕРГО:

Заяву на погодження акту прийняття в дослідну експлуатацію та акт прийняття в дослідну експлуатацію.

Зразок заповнення заяви на погодження акту прийняття в дослідну експлуатацію.

Бланк Акту вводу в дослідну експлуатацію

Зразок заповнення акту вводу в дослідну експлуатацію

ЛУЗОД/АСКОЕ клієнта вводиться в дослідну експлуатацію за умов:

клієнтом (проектною організацією) розроблено та погоджено з КИЇВЕНЕРГО проектну документацію на ЛУЗОД/АСКОЕ (технічне завдання, проект);

типи та місця встановлення приладів обліку, параметри вимірювальних трансформаторів відповідають погодженій проектній документації (технічне завдання, проект);

КИЇВЕНЕРГО має регламентований віддалений доступ до читання первинної бази даних лічильників.

Дослідна експлуатація встановленого обладнання проводиться протягом 3 місяців від дати його встановлення.

Протягом дослідної експлуатації ЛУЗОД/АСКОЕ клієнт (проектна або монтажна організація) мають провести державну метрологічну атестацію системи (п.3.14 ПКЕЕ). Такі послуги надає ДП "Укрметртестстандарт" або організація, яка діє від його імені. Копію свідоцтва про державну метрологічну атестацію ЛУЗОД/АСКОЕ клієнт має надати до центру обслуговування клієнтів КИЇВЕНЕРГО .

Протягом дослідної експлуатації ЛУЗОД/АСКОЕ клієнт має укласти додаткову угоду до договору про постачання електричної енергії «Щодо використання ЛУЗОД/АСКОЕ споживача». Для цього необхідно звернутися до центру обслуговування клієнтів КИЇВЕНЕРГО

У разі зміни місць встановлення розрахункового обліку клієнта, він протягом терміну дослідної експлуатації ЛУЗОД/АСКОЕ має внести відповідні зміни до договору про постачання електричної енергії. Для цього необхідно звернутися до центру обслуговування клієнтів КИЇВЕНЕРГО.

Після проходження державної метрологічної атестації ЛУЗОД/АСКОЕ клієнт (проектна або монтажна організація) складає акт прийняття в промислову експлуатацію.

Для введення обладнання в промислову експлуатацію клієнту потрібно надати до центру обслуговування клієнтів КИЇВЕНЕРГО:

Заяву на погодження акту прийняття в промислову експлуатацію

Зразок заповнення заяви на погодження акту прийняття в дослідну експлуатацію.

Акт прийняття в промислову експлуатацію ЛУЗОД/АСКОЕ

Бланк акту вводу в промислову експлуатацію

Бланк акту вводу в промислову експлуатацію автоматизованої системи комерційного обліку електроенергії

Зразок заповнення акту вводу в промислову експлуатацію

Зразок заповнення акту вводу в промислову експлуатацію автоматизованої системи комерційного обліку електроенергії

Свідоцтво про метрологічну атестацію

Ксерокопію акту введення в дослідну експлуатацію

ЛУЗОД/АСКОЕ клієнта вводиться в промислову експлуатацію за умов:

ЛУЗОД/АСКОЕ клієнта прийнята в дослідну експлуатацію (відповідний Акт погоджений КИЇВЕНЕРГО);

розрахункова схема клієнта за договором про електропостачання відповідає погодженій проектній документації (технічне завдання, проект);

проведена державна метрологічна атестація ЛУЗОД/АСКОЕ;

клієнтом укладено додаткову угоду до договору про постачання електричної енергії «Щодо використання ЛУЗОД/АСКОЕ Споживача»;

КИЇВЕНЕРГО має сталий регламентований віддалений доступ до читання первинної бази даних лічильників протягом терміну дослідної експлуатації ЛУЗОД /АСКОЕ клієнта.

Прийняті в промислову експлуатацію ЛУЗОД/АСКОЕ використовуються для проведення комерційних розрахунків з клієнтами.”[1]



Рисунок 1.1 Схема для підключення абонента до ЛУЗОД/АСКОЕ

З поданого вище, проаналізувавши всю надану інформацію на сайті «КиївЕнерго» можна зробити висновок, що самому обладнати свою оселю такою системою просто неможливо. Навіть юридичні аспекти у впровадженні будь-якої з вищенаведених систем є настільки складними, що середньостатистичний українець просто не захоче займатись цим питанням. Звідси виникають певні компанії, які за фінансову винагороду готові допомогти у встановленні та отриманні всіх дозволів, ліцензії, довідок, тощо для того аби спростити життя звичайному користувачеві.

Але все не так просто. Є ще певні фінансові питання, які будуть розглянуті трошки нижче. Також хотілося би зазначити, що жодна обленерго та Київенерго не зацікавлені в обладнанні цими системами звичайних осель. Максимум, на що може поступитись постачальник електроенергії – це обладнати цілий будинок. Але в реальності це доволі важко виконати.

Основною перевагою АСКОВ є те, що це вже готова мережа, яка, дійсно працює по всій Україні з повним моніторингом, аналізом, СМС-сповіщенням, тощо.

1.2 Пропозиції на ринку

Тепер пропоную приземлитись в наші реалії: а саме врахувати всі складності з юридичними моментами. Зрозуміло стає, що для під'єднання до цієї мережі необхідно скористуватись послугами компанії, яка вже має певний досвід у впровадженні таких систем. Для прикладу було знайдено за допомогою пошукової системи «Google» декілька таких компаній. Проаналізувавши інформацію подану на сайтах цих компаній було обрано для порівняння інжинірингову компанію «Вольт Енерго», оскільки

вони обладнали системами АСКОВ та Луод більше ніж 200 об'єктів. Також ця компанія знає і вміє як впровадити ці системи у звичайні оселі.

А тепер, увага, вартість: тільки за програмне забезпечення ця компанія просить 4704 гривні станом на 27.11.2019 для лише двох лічильників.

Враховуючи те, що в звичайній оселі є декілька лічильників:

Електроенергія

Холодне водопостачання

Гаряче водопостачання

Лічильник на систему опалення

Газовий лічильник

Треба купляти інше програмне забезпечення, яке буде розраховане на більшу кількість енергоносіїв. Таким чином необхідно буде витрати аж 9408 гривень (дані взяті на сайті компанії [2])

Але і це ще далеко не все. До уваги коротенький огляд системи:

Про продукт:

“Інжинірингова компанія «Вольт Енерго» представляє і рекомендує.

Програмний комплекс для автоматизованого обліку енергоресурсів (АСОВ) "енергоцентр" -використовується для організації комерційного і технічного обліку практично всіх видів енергоресурсів. Це і електроенергія, і тепло, і вода, і газ, і тепло.

Для оперативного контролю витрат на енергоресурси вона може бути синхронізована з системами управління підприємством і фінансовими програмами.

Система має наступні можливості (докладніше на наступній закладці на сайті Вольт Енерго):

враховує споживання енергоресурсів, шляхом збору даних по кожній точці (групі) обліку з заданим періодом контролю;

функціонує в будь-якому (автоматичному і ручному) режимі;

має можливість зберігати всі розрахункові параметри в базі даних;

забезпечує багатотарифний облік електроенергії;

дозволяє контролювати ліміти енергоспоживання;

інтерфейс програмного забезпечення ПК «Енергоцентр» дозволяє відображати інформації по системі АСКОЕ на моніторах автоматизованих робочих місць (АРМ) в графічному вигляді і табличному вигляді;

є можливість експорту даних в Excel і виведення на друк

забезпечує щоденний контроль працездатності приладів обліку;

система вільно взаємодіє з іншими подібними системами, АСКОЕ верхнього рівня енергокомпаній і системами управління підприємством.

Додатково може встановлюватися WEB інтерфейс для перегляду даних і генерації звітної інформації на будь-якому комп'ютері в будь-якій точці світу, при наявності інтернету.

Компанія є офіційним інсталятором програмного комплексу ПК «енергоцентр» для систем обліку АСКОЕ (АСКОЕ), асте (АСТОЕ). Перш ніж вибрати ПК «енергоцентр», як основну програму для впровадження систем АСКОЕ, нами були вивчено всі представлене на ринку ПЗ для даного сегмента. Основними критеріями, за якими вибирали ПК були:

- 1) Підтримка і оновлення програми

Програмний окомплекс ПК «енергоцентр» розроблений в Україні, відповідно, адаптований під українські реалії. При цьому відмінно взаємодіє з практично будь-яким програмним забезпеченням по диспетчеризації, технічному управлінню підприємством. Для цього над підтримкою ПК Енергоцентр постійно працює група програмістів та

інженерів, які вдосконалюють і оновлюють програму. Всі оновлення можна завантажити з сайту розробників. Там же можна побачити дату останніх оновлень. Таким чином, користувач ПК Енергоцентр завжди може бути впевнений, що програма «жива» і вона змінюється в залежності від зміни умов і кон'юктури на ринку енергоресурсів. Інжинірингова компанія Вольт Енерго, в свою чергу, є тестувальником ПК енергоцентр і одна з перших дізнається про якісь недоліки, доводячи інформацію до розробників для усунення цих недоліків.

2) Обмін даними в усіх прийнятих форматах з уже встановленими системами.

Незалежно від того яке ПЗ у Вас вже встановлено і в якому форматі Вам необхідно передавати і отримувати дані - ПК енергоцентр знайде «спільну мову» з вже існуючим ПЗ і забезпечить будь-який поширений формат передачі даних, що на енергопостачальні компанії, що на сервер користувача.

3) Можливість швидкого розширення - при необхідності знімати дані з нових (раніше не поширених на ринку України) приладів обліку енергоресурсів.

Якщо в діючу систему потрібно буде ввести новий прилад, який до того не був присутній на ринку України, то розробники протягом 2-х тижнів після заявки введуть пристрій в ПК «енергоцентр» в програмному режимі + 2 тижні на тестування і через місяць прилад буде спокійно читатися і дані з нього будуть відображатися в інтерфейсі (на моніторі) користувача

4) Досвід практичної роботи ПК «енергоцентр» на ринку обліку енергоресурсів.

На сьогоднішній день більше 400 програмних комплексів ПК енергоцентр встановлено і вже працює по всій Україні. З 2013 року ПК

енергоцентр почали встановлювати, як програму верхнього рівня, в деяких ОблЕнерго. Динаміка поширення ПК ЕНЕРГОЦЕНТР дуже позитивна. Це ще одне підтвердження працездатності та надійності інсталюється компанією Вольт Енерго програмного забезпечення.

5) Чіткість і доступність інтерфейсу для користувача.

Інтерфейс зрозумілий користувачеві з мінімальним знанням комп'ютера. Якщо користувач більш досвідчений, то можна елементарно налаштовувати під себе всі звіти

6) Ціна

З усіх представлених на ринку ПО для систем обліку енергоресурсів - у ПК Енергоцентр найдоступніша ціна з урахуванням безкоштовної підтримки. Градація цін залежить від кількості підключаються приладів обліку. Серйозне відміну від інших програм в тому, що у програми є можливість розширення. Тобто, якщо, наприклад, у Вас на підприємстві варто енергоцентр-10 (на 10 приладів обліку) і Вам потрібно розширити кількість приладів обліку в системі до 20, то Вам не потрібно купувати нове ПЗ на 20 лічильників (таким чином, «викидаючи» Енергоцентр-10), а потрібно доплатити різницю і Ваша програма буде оновлено до ПК енергоцентр-20.

Підтримувані лічильники

ПК АСКОВЕ «Енергоцентр», що інсталюється інжинірингової компанією Вольт Енерго, підтримує такі прилади обліку електроенергії.

Зарубіжні виробники:

Виробник: ITRON (Actaris) Франція - SL7000, ACE6000.

Виробник: EMH Metering Німеччина - LZQJ-XC, ITZ, ED2500.

Виробник: Elgama Літва- LZQM, EPQS

Виробник: Elster Росія - A1140, A1700, A1800, ЄвроАЛЬФА, Альфа.

Виробник: Landis & Gyr Швейцарія - ZMD, ZMG

Виробник: Iskra Словенія - ME172, MT174, MT372, ME372,
MT830

Виробник: Janitza Німеччина - UMG96S (Modbus)

Виробник: Инкотекс Росія - Меркурій 230

Українські виробники:

Виробник: ТОВ "НІК" - НІК (тарифні)

Виробник: "Телекард-прилад" - СТК (Енергія 9)

Виробник: ЕЛВІН - ЕТ

Виробник: ТОВ "TeleTec" [3]

А тепер хотілося би ще зазначити, що перелік підтримуваного обладнання є дуже малим. І звісно, що в переважній кількості населення встановлені не ці лічильники, що є в переліку підтримуваного обладнання.

Виходить, що користувачу для впровадження цієї системи необхідно поміняти лічильники, купити програмне забезпечення, а ще купити обладнання, яке є необхідним для цього програмного забезпечення.

1.3 Приблизна вартість проектів

Таблиця 1.1 Підсумок вартості готового комплексу для однієї оселі
(вже готовий варіант)

	Продукт	Вартість
ПЗ	Енергоцентр - 5	9408 грн.
Лічильник електроенергії	ACE 6000 DC, кл.т.1, 5(100)A, (A+, A-, R+, R-), моноготарифный, Itron (Actaris)	7020 грн.
Лічильник теплової енергії	НІК-7061-15-0-0-01,5, NiK	2700 грн.
Лічильник холодної води	НІК-7011Е-Х-15-0-0	855 грн.
Лічильник гарячої води	НІК-7011Е-Г-15-0-0	855 грн
Лічильник газу	Відсутній у перелічених виробників	_____
Ітого	Вартість за увесь комплект	20 838 грн

Тепер розглянемо вартість розробки, що буде запропонована у подальшій магістерській роботі.

Для обліку енергоносіїв пропонується використовувати наступне обладнання:

ESP32-CAM від AiThinker – 5шт. (на кожен лічильник)

Модуль Raspberry Pi 3 – 1шт. (на одну оселю)

Таблиця 1.2 Таблиця вартості для користувача розробленою системою

Прибор	Назва	Вартість
Прибор для фіксації даних	ESP32-CAM	205 грн. x 5 =1025
Модуль для збору, обробки та збереження інформації	Raspberry Pi 3	1 245 грн.
Блок живлення	Металлический внешний аккумулятор	24 грн. x 5 =120
Блок живлення для Raspberry	Блок живлення 5 В 2А з роз'ємом мікро-USB	97 грн.
Ітого	Уся система, кінцева вартість для користувача	2 487 грн.

Основні переваги системи, що буде розглянута в ході роботи:

- мала вартість;
- можливість використання з будь-якими лічильниками будь-яких виробників;
- автономність, як і вищезазначеної системи;
- гнучкість налаштування під потреби кожного користувача;

- свій міні-сервер збереження і обробки інформації стоїть прямо в користувача вдома;
- доступ до даних моніторингу з будь-якої точки земної кулі;
- енергоефективність;
- доступність.

1.4 Висновки з розділу 1

З оглядом на вищезазначені переваги питання доцільності розробки цієї системи відпадає автоматично. Звісно, що і користувач і розробник буде зацікавлений у впровадженні данної системи.

РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ ОБЛІКУ ЕНЕРГОНОСІЇВ

2.1 Вибір технології для передачі даних

Завдяки сучасним безпроводовим технологіям звичайний користувач мобільних та цифрових девайсів не пов'язаний дротами. Йому не потрібно носити на собі бухти кабелів, щоб отримати доступ до мережі інтернет, зателефонувати іншому абоненту чи передати якийсь файл на девайси інших користувачів. Бездротові технології оточують нас всюди та допомагають полегшити наше життя. Одже зупинимо свій погляд саме на безпроводній передачі даних, аби не тягнути до кожного лічильника кабелі.

Безпроводовою передачею називається ціла група технологій, умовно об'єднана одною гарною властивістю: відсутністю необхідності підключення дротів для передачі інформації на певні дистанції від одного девайса до іншого. Їх використовує увесь інтернет речей: сьогодні, навіть, мінімальна мережа не зможе без них працювати.

Класифікуються безпроводові технології за різними характеристиками. У щоденному вжитку зазвичай використовується їх маркетингові параметри, а не основні назви. Методи передачі інформації використовуються, теж, різні: від оптики до радіохвиль.

Найрозповсюдженіші види безпроводових технологій за територією охоплення сигналу :

WPAN. Персональні мережі, Bluetooth, ZigBee.

WLAN. Локальні мережі, Wi-Fi.

WMAN. Мережі міського масштабу, WiMAX.

WWAN. Самий глобальний різновид мереж, GPS, EDGE, HSPA і інші.

Беспроводові технології як середовище передачі даних використовують оточуючий простір замість кабелю. Але при цьому вони забезпечують користувачу високу мобільність завдяки широкому (в залежності від пристрою та типу) радіусу дії. Ці технології активно розширюються, швидкість передачі даних зростає, стабільність сигналу підвищується, енергетичні витрати знижуються. Потреба в розвитку бездротових технологій завдяки швидкому розповсюдженні в нашому житті переносних і мобільних не знижується.

Персональні та локальні безпроводові мережі:

Технології WPAN і WLAN — це те, що є в нас на щоденній основі. З цими технологіями працюють ноутбуки, мобільні пристрої, бездротові аксесуари, гаджети розумного будинку.



Рисунок 2.1 Архітектура WLAN

Мережі WPAN працюють на меншому радіусі, який може складати лише від 10 сантиметрів, так і до кількох метрів. Яскраві приклади — Bluetooth і ZigBee. Ці мережі зроблені для локального з'єднання пристроїв між собою та для передачі інформації між ними. Тому робимо висновок, що ця технологія автоматично відпадає, оскільки відстань від пристрою до пристрою може бути у декілька десятків метрів, додамо сюди ще й наявність завад і перешкод.

Мережі WLAN умовно масштабніші, хоча тежж локальні. Звичайний домашній Wi-Fi — це мережа WLAN, яка базується на стандарті IEEE 802.11. Слово Wi-Fi ніяк не розшифровується, від початкової назви, від якої склалась аббревіатура, вже давно відмовилися. Мережі типу WLAN працюють з більшим радіусом, ніж мережі типу WPAN. Швидкість передачі ними набагато вище, а також і захищеність набагато краще. Ось ця технологія набагато краще задовольняє потребам, оскільки ми матимемо не тільки покритою всю оселю цією мережею, але ще й матимемо доступ до глобальної мережі. Плюсом є й те, що ця технологія присутня у кожній оселі у вигляді звичного нам Wi-Fi.

Безпроводові мережі міського та глобального масштабу:

Мережі WMAN і WWAN здатні охоплювати найбільш великі території та локації. WMAN покривають місто, але WWAN можуть поширюватися на ще більший простір. А з ними ми постійно стикаємось в повсякденності.

Мережі WMAN, наприклад WiMAX, можуть охоплювати ціле місто. Це, по факту, той же Wi-Fi, тільки розтягнутий. WiMAX дає широкосмугове з'єднання для безлічі девайсів на великі відстані. Працює за схожою технологією, що і Wi-Fi, але в рази могутніше. Нажаль обладнання для цієї технології є значно дорожчим і складнішим ніж для технології WLAN. Тим паче той самий WiMAX не є таким розповсюдженим як Wi-Fi

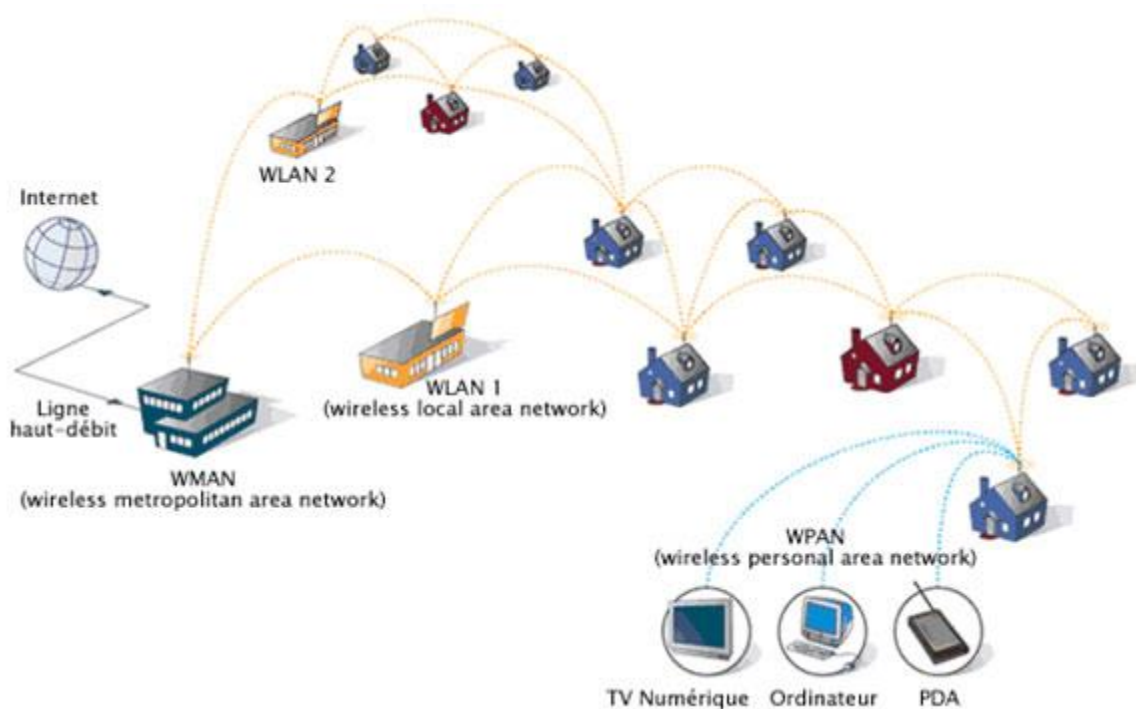


Рисунок 2.2 Архітектура WWAN

Мережі WWAN — це GPS, GPRS, GSM, EDGE, 3G і багато інших. Коли абонент здійснює дзвінки з мобільного телефону або виходить зі смартфона в мережу не по Wi-Fi, він працює з мережами типу WWAN. Працюють мережі за принципами пакетної передачі даних або комутації каналів. Дуже часто непогано шифруються, але захист в них далеко не максимальний. Нажаль маючи таку глобальність захисти всі діри в безпеці, мабуть, просто неможливо. Ця технологія є дуже потужним механізмом, має величезні можливості, але є і недоліки: такі як потреба в постійній абонентській платі за доступ до мережі інтернет, дуже часто відсутній зв'язок в місцях встановлення лічильників (підвали, шахти, тощо), а це обумовлює встановлення додаткового обладнання у вигляді репітерів та ретрансляторів GSM сигналу, що є доволі дорогим задоволенням.

З огляду на вищенаведені технології для розробки системи автоматизації обліку енергоносіїв була обрана технологія, що є, мабуть,

одной з найрозповсюдженіших та найпростіших в світі: WLAN. Саме ця технологія дозволить мати стабільне з'єднання, навіть в самих недоступних місцях оселі, оскільки зачасту відстань від передавача до приймача не буде перевищувати і 20 метрів. Також неоспоримим плюсом є те, що модуль Wi-Fi присутній майже на кожній сучасній платі, що в подальшому дозволить зменшити не тільки вартість проекту але і час витрачений на встановлення, розробку, програмування модулів і системи в цілому.

2.2 Вибір обладнання для проекту

На сьогоднішній день на ринку представлено величезне різноманіття плат і компонентів, які мають вбудовані мікропроцесори і інші модулі, такі як плата Wi-Fi, роз'єми вводу-виводу інформації, як цифрової так і аналогової. Але особняком серед них всіх стоїть сімейство Arduino із зручним середовищем для програмування ArduinoIDE

“Як каже Вікіпедія: «Arduino (Ардуіно) — апаратна обчислювальна платформа для аматорського конструювання, основними компонентами якої є плата мікроконтролера з елементами вводу/виводу та середовище розробки Processing/Wiring на мові програмування, що є підмножиною C/C++». Людськими словами — це дешева плата яка є у вільному доступі, яка сумісна і з Windows і з Mac та навіть з Linux, до якої можна підключити багато чого, від вирваної кнопки із старої мишки до LED екрану. І все це легко програмується навіть новачком. І це реально так, за власним досвідом скажу, що розібратися у цьому всьому не так вже й важко, я починав ще підлітком, не маючи ніякого досвіду в цьому, тепер же програмування Arduino та загалом програмування для мене є простою

справою, де можна розкрити не тільки свої математичні навички, а й весь свій творчий потенціал.

Суть створення Arduino-проектів проста, ви берете одну плату з великої лінійки, підключаєте до її контактів датчик чи якийсь модуль, і пишете, коли на ці контакти потрібно подавати струм чи наоборот перевіряти наявність струму на них. Ось простий приклад. До одного контакту ви підключили датчик світла, а до другого реле з лампою і пишете команди щоб програма перевіряла сигнал з сенсора і за умови його відсутності вмикала реле. Ставим цю систему на подвір'ї, і ось, коли вечоріє у нас вмикаються ліхтарі. І це лише найпростіший приклад, після невеликої практики Ви навчитеся створювати системи, які будуть вмикати освітлення в домі по хлопку або командою “Ok, Google”, які будуть перевіряти чи підліті квіти та в потрібний момент — підливати їх. Ці плати також вміють керувати двигунами, а отже можна створювати автономні машинки, роботів, балансуючих на двох колесах, приводи для електроштор та 3D принтери.

Для початку роботи потрібно розжитися одною з плат, вона може бути як оригінальною, так і аналогом, так і китайською копією або ж взагалі власноруч спаяною. У мене уже друга китайська копія, перша була UNO, а зараз MEGA і проблем у роботі ні з першою, ні з другою не було, а ціна в рази менша. Також потрібне буде програмне забезпечення, воно є на офіційному сайті у вільному доступі і з усіма вище перерахованими платами справно працює. Модулями та сенсорами ви розживетесь з часом, їх можна буде як купити, так і зібрати самому з підручного хламу.

І так, спочатку підготовка.

Крок 1. Визначитися з найбільш підходящою для проектів платою. Благо їхній вибір є досить великим, набагато більшим ніж на зображенні.

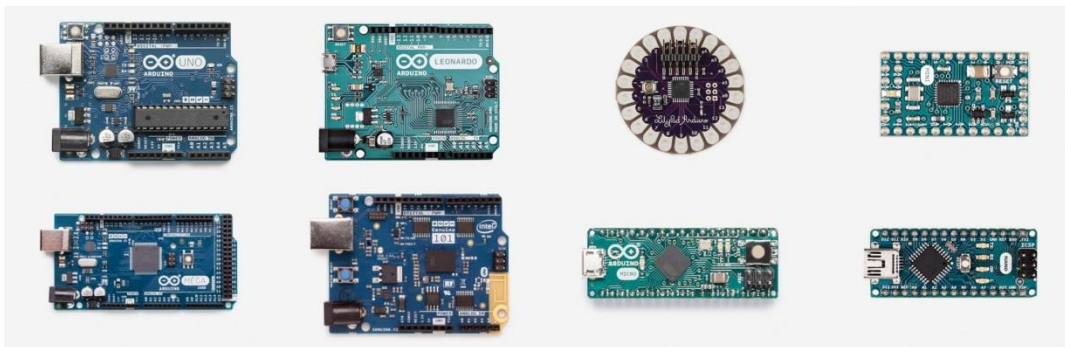


Рис 2.3 Плати Arduino

Для проектів носимих компактних гаджетів є Mini, Nano та Micro. Для чогось більшого ідеально підходить Mega. Особисто я рекомендую для початку брати UNO або трішки докласти та взяти MEGA, сміло можете брати копії, а не оригінали.

Крок 2. Потрібно встановити програмне забезпечення Arduino IDE. Воно є на офіційному сайті, натисніть у списку на вашу операційну систему.

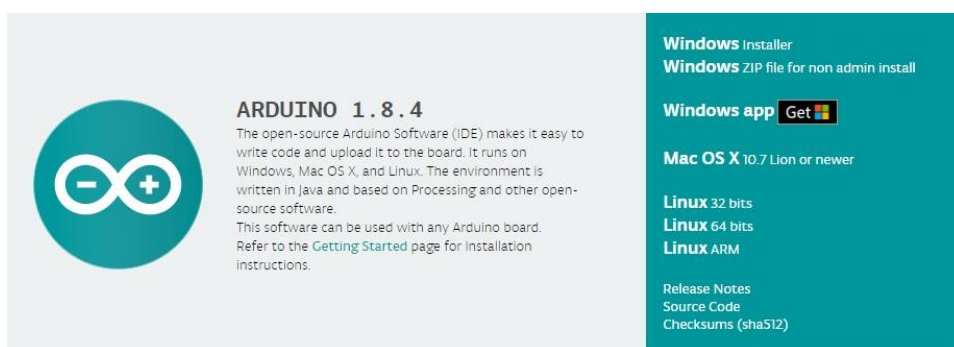


Рисунок 2.4 Середина програмування Arduino IDE – Завантаження

Крок 3. Налаштовуємо це програмне забезпечення(Arduino IDE). Підключаємо нашу плату до свого ПК або Mac. Відкриваємо Arduino IDE. У вкладці “Tools” наводимо курсор на “Board:” та натискаємо на плату, яку ви маєте (назва тієї плати, яку щойно підключили).

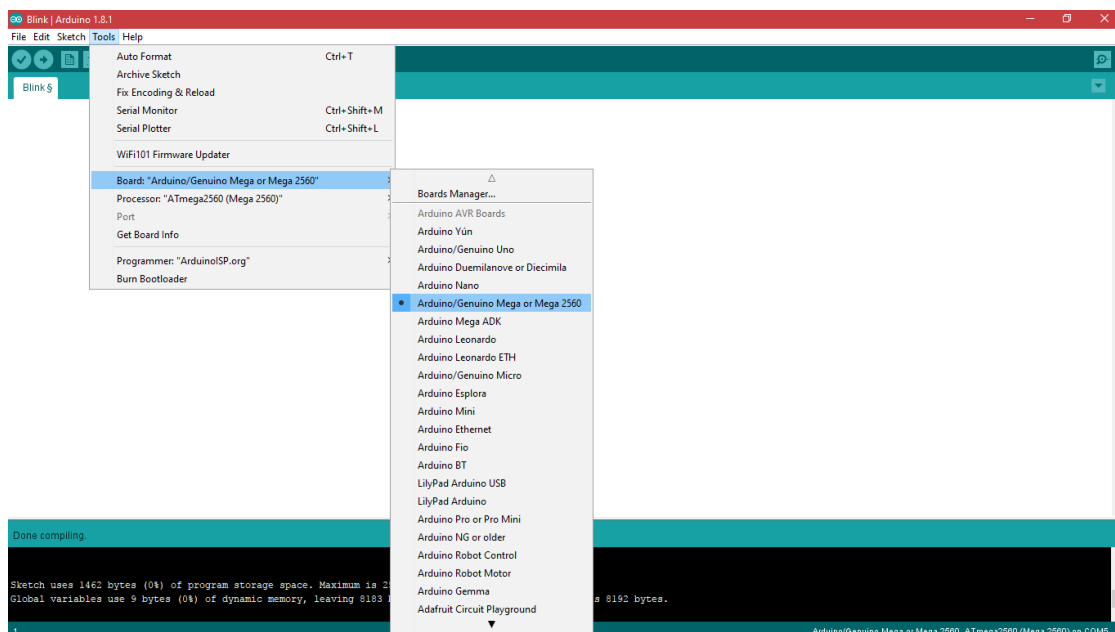


Рисунок 2.5 Вибір плати

Потім у тій же вкладці “Tools” наводимо на “Port:” та вибираємо той порт, до якого підключено Arduino. Скоріш за все програма це зробить за вас.

Ось і все: можна розпочинати створення своїх проектів” [4].

2.3 Вибір плати

Оскільки вже раніше в ході навчання та проходження практики на виробництві доводилось працювати з різними платами то в наявності на початок розробки була плата Arduino UNO R3.

Arduino Uno — це найбільш широко використовувана в навчанні плата мікроконтролерів та елементів з відкритим кодом на базі мікроконтролера ATmega328P. У склад плати входить все необхідне для зручної роботи з мікроконтролером:

- 6 аналогових входів,

- 14 цифрових входів/виходів (з них 6 можуть використовуватися в якості ШІМ-виходів),
- роз'єм USB, роз'єм живлення,
- кварцовий резонатор на 16 МГц,
- роз'єм для програмування всередині схеми (ICSP),
- кнопка скидання.

Для того аби розпочати роботу з пристроєм достатньо просто подати живлення від AC/DC-адаптера або живлячого елемента, або підключити його до персонального комп'ютера за допомогою USB-дрота.

На відміну від інших попередніх старих плат Arduino, Uno в якості перетворювача інтерфейсів USB-UART використовує мікроконтролер замість мікросхеми FTDI - ATmega16U2 (ATmega8U2 до версії R2).

На Arduino Uno R2 для спрощення ритуалу оновлення прошивки доданий резистор, що підключає до «землі» лінію HWB мікроконтролера 8U2.

Характеристики:

Таблиця 2.1 Характеристики Arduino UNO R3

Основні критерії	Значення
Мікроконтролер	ATmega328
Робоча напруга	5В
Напруга живлення (рекомендований)	7-12В
Напруга живлення (граничне)	6-20В
Цифрові входи / виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Максимальний струм одного виведення	40мА
Максимальний вихідний струм виводу 3.3V	50мА
Flash-пам'ять	32 КБ (ATmega328) з яких 0.5 КБ використовуються завантажувачем
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактова частота	16МГц

Живлення

Arduino Uno може живитися як від USB так і від зовнішнього будь-якого п'яти вольтового джерела живлення — тип джерела пристроєм вибирається автоматично.

В якості зовнішнього джерела живлення для плати (НЕ USB) може використовуватися мережевий акумулятор/батарея або AC/DC-адаптер. Штекер адаптера (центральний контакт — позитивний, діаметр — 2.1 мм) необхідно під'єднати до відповідного роз'єму живлення на платі. У разі живлення від акумулятора / батареї, її дроти необхідно під'єднати до виходів Gnd і Vin роз'єму POWER.

Вольтаж зовнішнього джерела живлення може лежати в межах від 6 до 20 В. Однак, якщо зменшувати напругу живлення нижче 7 В то це призведе до зменшення напруги на виході 5V, що призведе до нестабільної роботи пристрою. В той час використання напруги на джерелі живлення більше 12 В може призводити до перегріву стабілізатора напруги а також можливо і виходу плати з ладу. Одже рекомендується джерело живлення з напругою в діапазоні від 7 до 12 В.



Рисунок 2.6 Виводи Arduino Uno R3

Нижче перелічені виходи живлення, що знаходяться на платі:

- VIN. Напруга, що приходить в Arduino на пряму від зовнішнього джерела живлення (не пов'язане з 5 В від USB або іншої будь-якою стабілізованою напругою). Через цей вивід можна подавати зовнішнє

живлення і споживати струм, тоді коли пристрій буде живиться від зовнішнього адаптера.

- 5V. На вихід надходить напруга 5 В від стабілізатора напруги плати, в незалежності від того, як живиться пристрій: чи від адаптера (7 — 12 В), чи від USB (5 В), чи через вивід VIN (7 — 12 В). Живити пристрій через вивід 5 В або 3V3 просто не рекомендується, оскільки в цьому випадку не буде використовуватись стабілізатор напруги, що може призвести до виходу плати з ладу.

- 3V3. 3,3 В, надходять від стабілізатора напруги на платі. Максимальний струм, споживання від цього виводу, становить 50 мА.

- GND. Вивід землі.

- IOREF. Цей вихід надає іншим підключеним платам розширення інформацію про робочу напругу мікроконтролера Arduino. Залежно від напруги, на виході IOREF, плата розширення перемикається на відповідне джерело живлення або задіює перетворювачі рівнів, що дозволить їй працювати разом з 5 В, так і з 3,3 В-пристроями.

Входи і виходи:

З задіянням функцій `pinMode ()`, `digitalWrite ()` і `digitalRead ()` кожний з 14 цифрових виходів може працювати як вхід так і вихід. Рівень напруги на виходах обмежений на рівні 5 В. Максимальний струм, що може віддавати або споживати один вихід, становить 40 мА. Всі виходи з'єднані з внутрішніми резисторами (за замовчанням відключеними) опором 20-50 кОм. Окрім цього, деякі виходи Arduino можуть виконувати деякі додаткові функції:

Послідовні інтерфейси: виходи 0 (RX) і 1 (TX). Використовуються для отримання (RX) і передачі (TX) інформації по послідовним інтерфейсам. Ці виходи з'єднані з виходами мікросхеми ATmega8U2, яка є перетворювачем USB-UART.

ШІМ: виходи 3, 5, 6, 9, 10 і 11. Функція `analogWrite ()` може виводити 8-бітові аналогові показники в вигляді ШІМ-сигналу.

SPI: виходи 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Застосуванням бібліотек SPI дані виходи можуть здійснювати зв'язок інтерфейсом SPI.

Світлодіод: 13-ий пін. Вбудований світлодіод, підключений до виходу 13. При заданні значення HIGH світлодіод - включається, при LOW — вимикається.

Arduino Uno також має 6 аналогових входів (A0 — A5), кожний з них може прийняти аналогову напругу у вигляді 10-бітного числа (1024 рівня). За замовчанням, вимір напруги здійснюється відносно діапазону 0 - 5 В. Але, верхню межу цього діапазону можна замінити, використовуючи вихід AREF і функцію `analogReference ()`. Окрім цього, деякі з аналогових входів плати мають також додаткові функції:

- TWI: вихід A4 або SDA і вихід A5 або SCL. Використовуючи бібліотеку `Wire` дані виходи можуть здійснювати зв'язок за допомогою інтерфейсу TWI.

Окрім перелічених на платі існує ще декілька виходів:

- AREF. Опорна напруга для аналогових входів. Задіюється за допомогою функції `analogReference ()`.

- Reset. Формування низького рівня сигналу (LOW) на данному виході призведе до перегрузки мікроконтролера. У повсякденні цей вихід служить для роботи кнопки «скидання» на платах.

Пам'ять:

Обсяг флеш-пам'яті ATmega328 складає лише 32 КБ (з яких 0.5 КБ використовуються загрузчиком). Мікроконтролер ще має 2 КБ пам'яті SRAM і 1 КБ EEPROM (в якій можна зчитувати або записувати дані за допомогою бібліотеки EEPROM).

Зв'язок:

У ATmega328 є отримувач UART, який дозволяє здійснювати послідовний зв'язок завдяки цифровим виходам 0 (RX) і 1 (TX). Мікроконтролер ATmega16U2 на платі є зв'язок цього приймача з USB-роз'ємом комп'ютера, та при під'єднанні до ПК дозволяє Arduino з'явитися в переліку обладнання як віртуальний COM-порт. Прошивка мікросхеми 16U2 підтягує стандартні драйвера USB-COM, тому установка інших нестандартних драйверів не потрібна. На операційній системі Windows необхідний тільки .inf-файл. До пакета програмного забезпечення Arduino входить спеціальна програма, яка зчитує і відправляє на Arduino прості текстові дані. При трансляції даних через мікросхему-перетворювач USB-UART від комп'ютера, на платі будуть блимати світлодіоди RX і TX. (При послідовній передачі інформації за допомогою виходів 0 і 1, без допомоги USB-перетворювача, ці світлодіоди задіюються).

Бібліотека SoftwareSerial може реалізувати послідовний зв'язок на будь-яких цифрових виходах Arduino Uno.

Також на мікроконтролері ATmega328 реалізована підтримка послідовних інтерфейсів SPI і I2C (TWI). У ПЗ Arduino входить бібліотека Wire, що також може спростити роботу з шиною I2C; для отримання більш детальної інформації дивіться документацію. Для роботи з інтерфейсом SPI є бібліотека SPI.

Захист USB від перевантажень:

Не дивлячись на те, що більшість комп'ютерів мають особистий захист, такі запобіжники дають додатковий рівень захисту. Якщо раптом на USB-порті споживається струм більше ніж 500 мА, запобіжник

автоматично роз'єднанає сесію до усунення причин короткого замикання або перегрузки.

Фізичні характеристики:

Найбільша довжина і ширина друкованої плати Arduino Uno складає лише 6,9 см і 5,4 см, з врахуванням роз'єму USB і роз'єму живлення, які виступають за фізичні межі плати. Чотири отвори для фіксації дозволяють зафіксувати плату до якоїсь поверхні або корпусу. Але, відстань між цифровими виводами 7 і 8 не кратне звичним 2,54 мм і складає 4 мм.

І все здавалося би непогано, за виключенням того, що ця плата не має ні вбудованої камери, ні модуля Wi-Fi. Але до неї можна підключити всі ці модулі. Таке рішення і було прийнято на початку розробки системи. Але, як показала практика – це дуже не практично. Ось приклад з'єднання плати Arduino з одною з найпростіших камер на ринку OV7670.

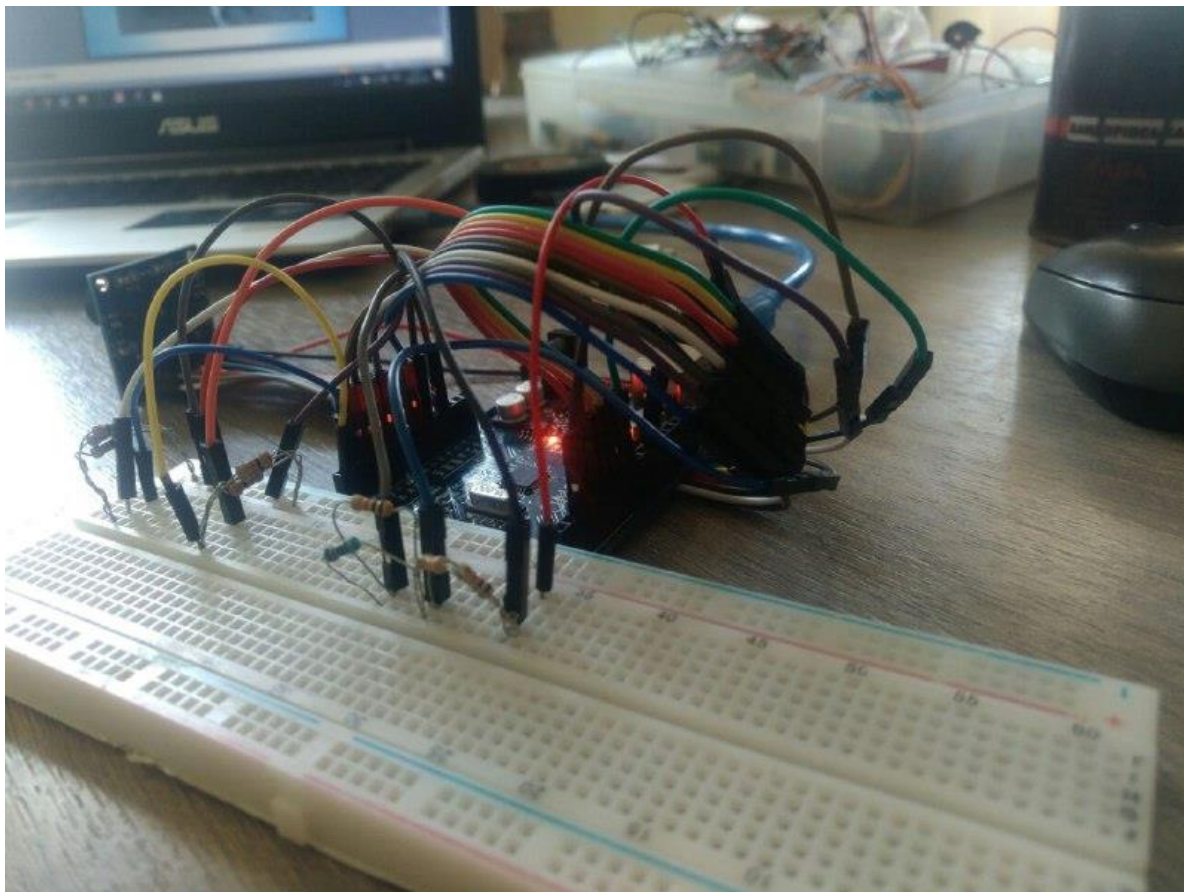


Рисунок 2.7 Зразок підключення камери до Arduino

Як видно з фотографії кількість дротів і з'єднань тільки для підключення камери вже є захмарною, а нам ще треба підключити плату для передачі даних по мережі WLAN. Можна тільки уявити який би хаос з дротів там творився.

Як наслідок в ході польових випробувань, фото буде додано нижче ні про яку стабільність передачі зображення мови йти й не могло. І все це стало наслідком величезної купи з'єднань.

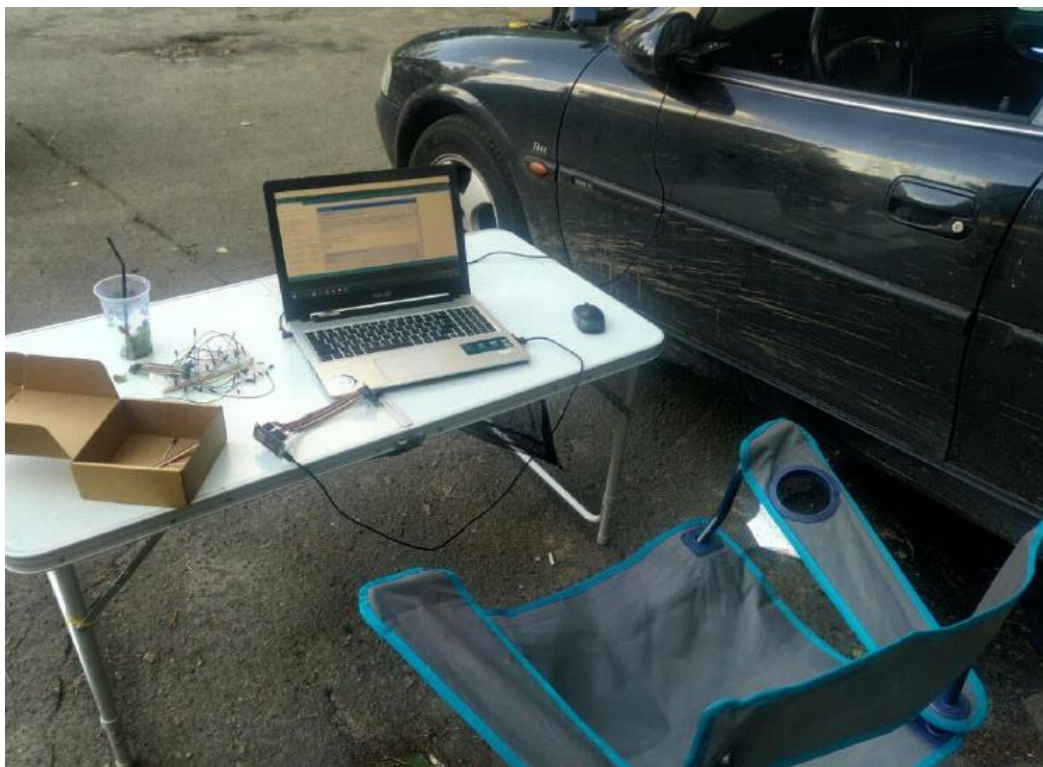


Рисунок 2.8. Польові випробування першого зразка

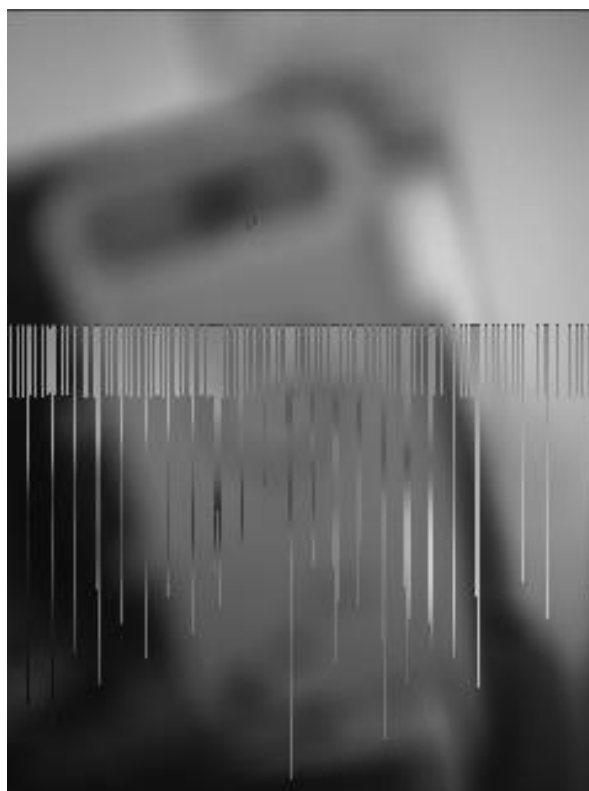


Рисунок 2.9 Зразок нестабільної передачі зображення з камери

Після довгих вдосконалень, аналізу, спрощення ,покращення з'єднань та переробки програмного забезпечення аж до рівня заліза було отримано більш стабільне з'єднання, але це, нажаль, не допомогло, адже питання під'єднання до мережі інтернет залишилось відкритим. А роз'ємів на платі Arduino UNO R3 вже просто не вистачало. Та і в перспективі би хотілось під'єднати по декілька камер до плати аби зчитувати інформацію з декількох лічильників.

Також в ході роботи були прийнята спроба обробити фотографію з лічильника безпосередньо на самій платі Arduino UNO R3. В наслідок цього експерименту було втрачено роботоспособність камери та плати і довелось обдумувати інші варіанти для цього проекту.

Тепер, маючи на руках вже певну кількість зіпсованої техніки не хотілось би витрачати час і кошти на подальші спроби реалізації проекту. Настав час все детально продумати. Проаналізувавши ринок, перечитавши купу профільних форумів увагу привернула плата ESP32-CAM від виробника AiThinker.

Про сам модуль:

“ESP32-CAM - плата для розробки ESP32 і камера OV2640 2MP.

Потужний двоядерний 32 бітний процесор Xtensa з частотою 160 МГц.

Застосовується в якості камери спостереження. Систем розпізнавання осіб, жестів, QR кодів.

Дана версія модуля має розпаювання дроселя на свою внутрішню антену.

(Для використання зовнішньої антени, дросель необхідно перепаять.)

Технічні характеристики:

WiFi 802.11b / g / n

Bluetooth 4.2 LE з друкованою антеною, роз'єм u.FL. (IPX13)

32 Мбіт SPI флеш-пам'ять

4 Мбіт PSRAM

Слот для мікро SD-карти до 4 Гб

Габаритні розміри - 40.5 x 27 x 4.5 мм

Вага - 10 гр.

Камера:

роз'єм FPC

Підтримка камер OV2640, OV7670

Формат зображення - JPEG (підтримує тільки OV2640), BMP,
відтінки сірого

Світлодіодна підсвітка

Розширення - UART, SPI, I2C, PWM

кнопка скидання

Живлення - 5 вольт через контактний роз'єм

Споживана потужність:

Світлодіодне підсвічування вимкнено - 180 мА @ 5 В

Світлодіодне підсвічування включено (макс. Яскравість) - 310 мА
@ 5 В

Deep-sleep - 6 мА @ 5 В мінімум.

Modem-sleep - 20 мА @ 5 В мінімум.

Light-sleep - 6.7 мА @ 5 В мінімум” [5].

Як бачимо ця плата є доволі потужним агрегатом для створення зображення, до того ж має світлодіодну підсвітку, що було би необхідним

в момент фотографування лічильника, оскільки вони зазвичай встановлені в погано освітлених місцях. Також величезною перевагою є її малі витрати енергії, так наприклад в режимі глибокого сну лише 6мА.

Але, нажаль, потужності цієї плати навряд чи вистачить для обробки отриманого зображення. Тому для обробки зображення гарно би було використовувати сервер. І тут є вирішення цієї задачі, а саме Raspberry Pi 3. Під час проходження практики ще у бакалавраті неодноразово доводилось стикатися і працювати з цією платою. Вона по факту є повноцінним комп'ютером з повноцінним графічним інтерфейсом, що є дуже зручним для роботи рядового користувача.

“Raspberry Pi 3 Model B - цей міні-комп'ютер отримав більш потужний просунутий процесор і бездротові пристрої комунікації. Нова версія Raspberry Pi працює набагато швидше попередників з такими додатками як електронні таблиці, обробка текстів, відтворення відео високої чіткості і гри. Raspberri PI 3 може працювати під управлінням декількох різновидів Linux і навіть безкоштовної Windows 10.

У новій моделі "малини" використовується потужний процесор Broadcom BCM2837 ARM Cortex-A53 64-бітний процесор Quad-Core SoC. Графічний процесор забезпечує підтримку Open GL ES 2.0, апаратне прискорення OpenVG і 1080p30 H.264 високого рівня декодування і здатний обробити до 1Gpixel / с, 1.5Gtexel / с або 24 GFLOPS інших обчислювальних ресурсів. Що це означає? Це означає, що якщо ви підключите Raspberry Pi 3 до вашого HDTV телевізору Ви можете переглядати з якістю BluRay відео, використовуючи H.264 на 40МБіт / с.

Найбільше зміна, яка була зроблена в Raspberry Pi 3, є оновленням основного процесора наступного покоління і поліпшені можливості комунікації з Bluetooth Low Energy (BLE) і BCM43143 WiFi контролером на борту. Крім того, в Raspberry Pi 3 розробники поліпшили управління

живленням і підвищили потужність USB до 6Вт (1,2 ампер) для підтримки більш потужних зовнішніх USB пристроїв. Чотири вбудованих USB порту Raspberry Pi 3 забезпечують широкі можливості для підключення миші, клавіатури, або будь-який інший периферії, але якщо ви захочете додати ще більше USB пристроїв, то потрібно буде використовувати USB концентратор. Майте на увазі, що рекомендується використовувати хаб, щоб не перевантажувати стабілізатор напруги на платі. Включення Raspberry Pi 3 легке і просте - досить підключити будь-який USB джерело живлення в порт мікро-USB. На платі немає кнопки живлення, так що Pi почне завантажуватися, як тільки підключений до джерела живлення. Щоб вимкнути його просто вимкніть живлення. Чотири вбудованих USB-порту можуть виводити до 1.2А, дозволяючи підключати більше ненажерливі пристрої USB (Це вимагає 2,5А мікро-USB джерело живлення).

Додатково до всього є периферійні пристрої низького рівня, що робить Raspberry Pi відмінним інструментом для збору інформації і управління виконавчими пристроями. 40-контактний роз'єм GPIO на Pi дає Вам доступ до 27 GPIO, UART, I2C, SPI, а також 3,3 і 5V джерела живлення. Кожен контакт GPIO ідентичний своєму попереднику Model B+.

Характеристики:

Процесор: Broadcom BCM2837 64bit ARM Cortex-A53 Quad Core
SoC

Робоча частота процесора: 1.2ГГц

Оперативна пам'ять: 1 ГБ

Пам'ять програм і даних: MicroSD

USB порти: 4 x USB 2.0 порту з виходом до 1.2 А

Розширений 40-контактний роз'єм GPIO

Відео / аудіо вихід через 4-контактний роз'єм 3.5 мм, HDMI, CSI камери або Raw LCD (DSI)

Мережевий інтерфейс: 10/100 Ethernet (RJ45)

Бездротовий інтерфейс: BCM43143 WiFi і Bluetooth Low Energy (BLE)

Периферійні пристрої низького рівня:

27 x GPIO

UART

I2C інтерфейс

SPI шина з двома сигналами вибору мікросхем

+ 3.3V

+ 5V

GND

Вимоги до живлення: 5V @ 2.4 А через джерело живлення MicroUSB

Розміри: 85 x 56 x 17 мм

Підтримка: Raspbian, Windows 10 IoT Core, OpenELEC, OSMC, Pidora, Arch Linux, RISC OS і багато іншого” [6].

2.3 Архітектура проекту

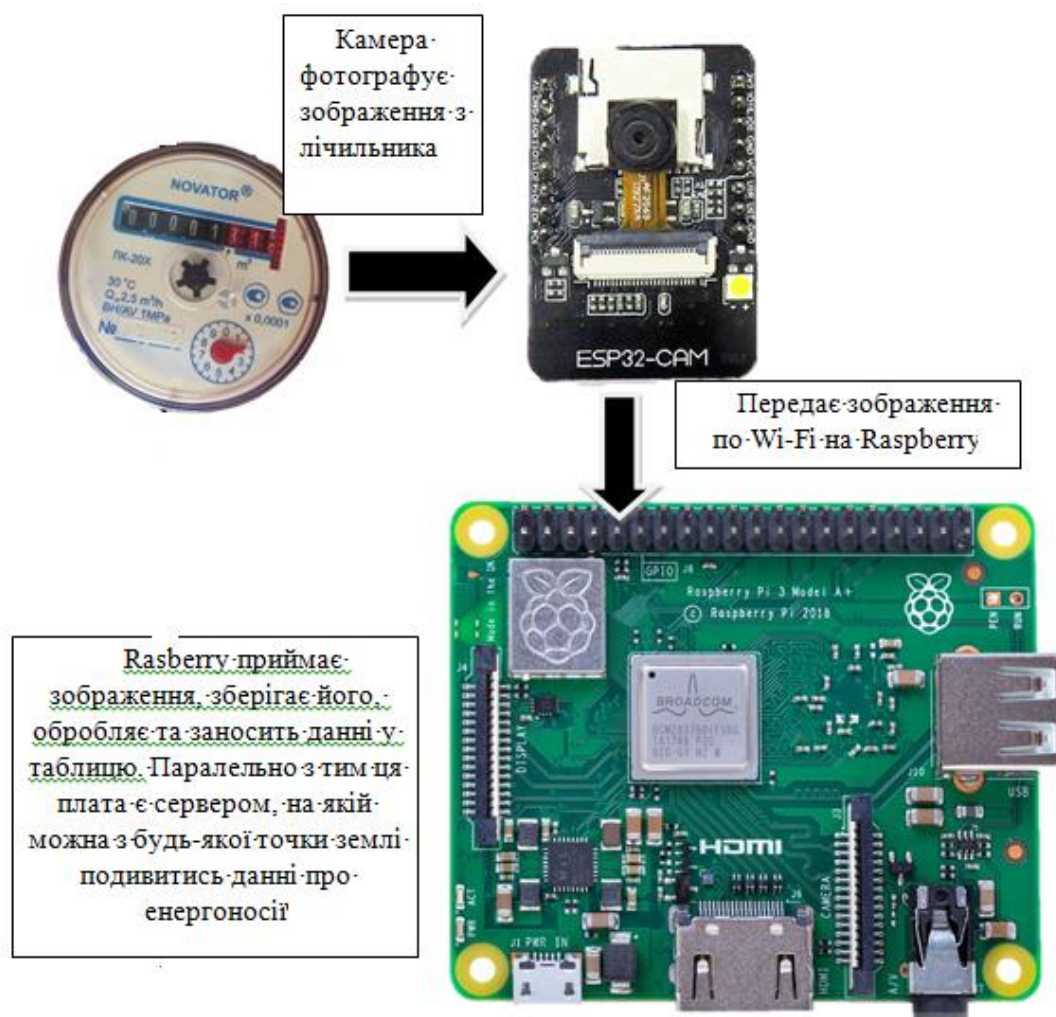


Рисунок 2.10 Архітектура розробленого проекту.

За замовчанням плата ESP-32CAM знаходиться у режимі глибокого сну. Режим глибокого сну має дуже мале використання енергоресурсів. За командою, яка поступає від спеціального малопотужного, але енергоефективного процесора вона прокидається раз на чотири години і починає виконувати скрипт:

Прокинутись

Під'єднатись до мережі Wi-Fi

Встановити з'єднання з сервером, який є на Raspberry
 Зробити фотографію із спалахом і зберегти її в оперативну пам'ять
 Відправити фотографію на сервер
 Перехід в режим глибокого сну

Ця плата має на вибір декілька варіантів живлення: це може бути, як звичайний блок живлення на 5В і 500мА, так і PowerBank у випадку, якщо не має можливості під'єднатися до мережі 220В. Враховуючи те, що споживання плати у режимі глибокого сну 6 мА то середньостатистичного PowerBank на 10 000мА/год вистачить на 69 діб. Але також, ще треба врахувати, що пристрій має ще робити фотографії та відправляти їх на сервер з певним інтервалом часу. Тому краще би було під'єднати до плати блок живлення, тим паче, зазвичай лічильники холодної та гарячої води стоять поруч і це б дозволило зекономити на елементах живлення. З приводу газового лічильника: завжди біля лічильника має знаходитись пристрій відстежування газів. Він працює від мережі 220 В. Що стосується лічильника електроенергії – то тут, навіть і не виникає питання, де взяти живлення.

Також треба зазначити, що камера має бути встановлена таким чином аби при фотографуванні зі спалахом було розбірливо видно усі символи, які є на лічильнику.

Запрограмувати цю плату дуже легко. Для цього необхідно встановити на комп'ютер середу розробки програмного забезпечення ArduinoIDE та залити скрипт виконання. Для того аби з'єднати комп'ютер з платою необхідно мати спеціальний перехідник-програматор USB 2.0 в TTL UART.



Рисунок 2.11 Програматор і ESP32-CAM

Після того як плата буде запрограмована її можна підключити до живлення і вона готова до роботи.

Для того аби розпочати роботу з Raspberry необхідно записати на карту пам'яті образ з операційною системою: Raspberrian OS.

Після цього можна під'єднувати до малини живлення 5В 2,5А. Також для налаштування не забуваємо, що необхідно під'єднати монітор по HDMI кабелю, USB-клавіатуру та мишку. Інакше налаштування доведеться виконувати по віддаленому з'єднанню з комп'ютеру, що не зручно та не продуктивно.

Після запуску починаємо налаштування.

Алгоритм роботи Raspberry:

Запуск операційної системи

Під'єднання до існуючої мережі Wi-Fi

Запуск серверу

Відкриття портів для прийому файлів

Включення режиму прослуховування портів

Прийом фотографії та збереження її від ESP32-CAM (може бути декілька камер)

Обробка фотографії та збереження тексту у раніше створену таблицю

Перехід в режим прослуховування портів

Сама малина є дуже потужним інструментом. По факту це є потужний комп'ютер заключний у маленьку плату із повноцінним інтерфейсом. Прикладом того наскільки малина є потужною є той факт, що в 2018 році сервери НАСА були взламани за допомогою Raspberry PI 3.

Але малина є дуже вибагливою до живлення: необхідно мати блок живлення потужністю у 2,5А і напругою у 5В. Зрозуміло, що ні про які PowerBank-и не може і йти мови. Тому рекомендовано встановити малину десь біля телевізора з HDMI входом та розетки для забезпечення максимально оперативного втручання в разі потреби та неперервного живлення. Взагалі, хотілося б додати, що малину можна паралельно використовувати як свій особистий сервер для зберігання даних, або в разі, якщо малина підключена до телевізора – можна використовувати її в якості цифрової приставки для програвання медіа-контенту та навіть ігор. І паралельно з цим вона буде виконувати свої обов'язки по збору, обробці та збереженню інформації

2.4 Висновки з розділу 2

В цьому розділі було розглянуто декілька технологій для передачі даних, була обрана оптимальна – WLAN. Також експериментальним

шляхом було встановлено не відповідність до потреб помилково вибраної плати. Потім все-таки була коректно обрана плата, яка задовольняє усім потребам і має запас потужності, який за необхідності можна буде використати в майбутньому. Також на цьому обладнанні було продумано архітектуру проекту.

РОЗДІЛ 3 МАТЕРІАЛЬНА БАЗА

3.1 Raspberry

Плата Распбері є доволі потужним інструментом в роботі. Але вона має один величезний недолік – відсутність корпусу. Це робить її непридатною до експлуатації в реальному житті. Тому для її захисту був надрукован на 3D-принтері корпус, який має пасивне охолодження.



Рисунок 3.1 Вид на Raspberry здаду



Рисунок 3.2 Вид на Raspberry зпереду

3.2 Корпус для лічильника і передатчика (ESP32-CAM)

Для того аби якимось чином зафіксувати камеру відносно лічильник та захистити камеру і скло лічильника від зовнішніх факторів довелось розробити, прокреслити і надрукувати цілу систему, що дозволяє оперативно змінювати соосність камери з лічильником, відстань від камери до лічильника, розміщення всього корпусу на лічильнику, тощо.

Спочатку для фіксування камери та для того аби закрити зайву інформацію з корпусу лічильника була розроблена спеціальна кришка (Рисунок 3.4)

Для того аби можна було в подальшому відцентрувати камеру відносно вікна та виставити відстань була розроблена спеціальна обойма по якій вгор-вниз та навколо своїй осі могло рухатись кріплення камери із самою камерою. (Рисунок3.5)

Для позиціонування камери в просторі відповідно було прокреслено і надруковано спеціальне кріплення із вбудованою лінзою. Лінза є необхідною для того аби змінити фокусну відстань, оскільки камера має фокус 1,8м. а в проекті потрібно 2-7 см. (Рисунок 3.6)

Для того аби ESP-32CAM надійно зафіксувати в кріпленні зверху встановлюється фіксатор (Рис 3.7)

Відповідно, зібравши цю складну конструкцію отримуємо готовий прилад (Рисунок 3.8) На фото представлений із підключеним діодом, виведеними дротами живлення та двома дротами для налаштування.



Рисунок 3.3 Лічильник

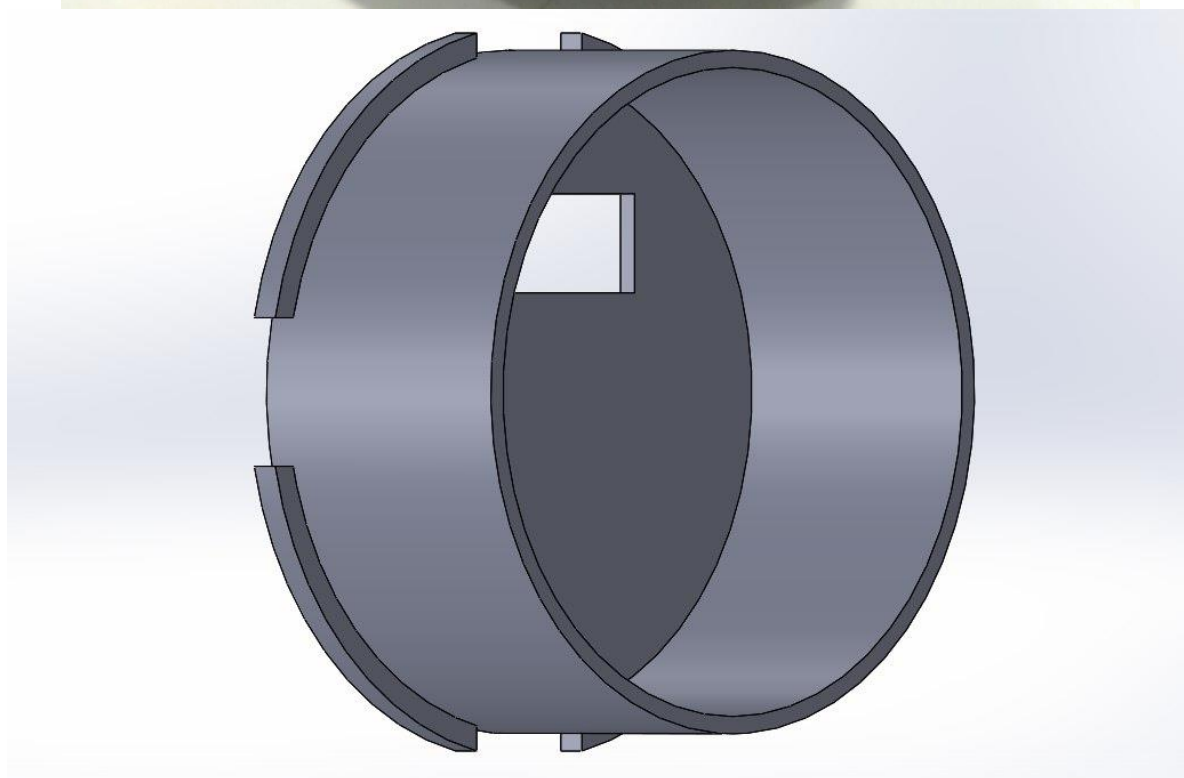


Рисунок 3.4 Кришка лічильника

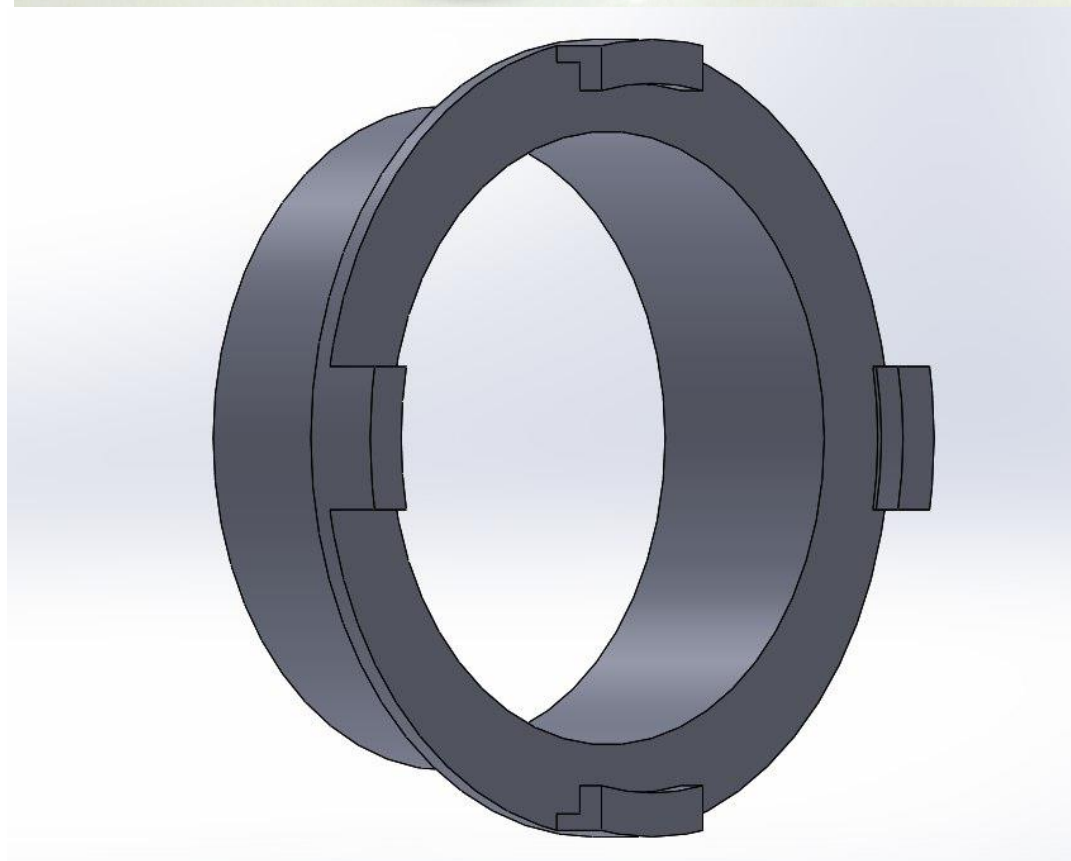


Рисунок 3.5 Обойма лічильника

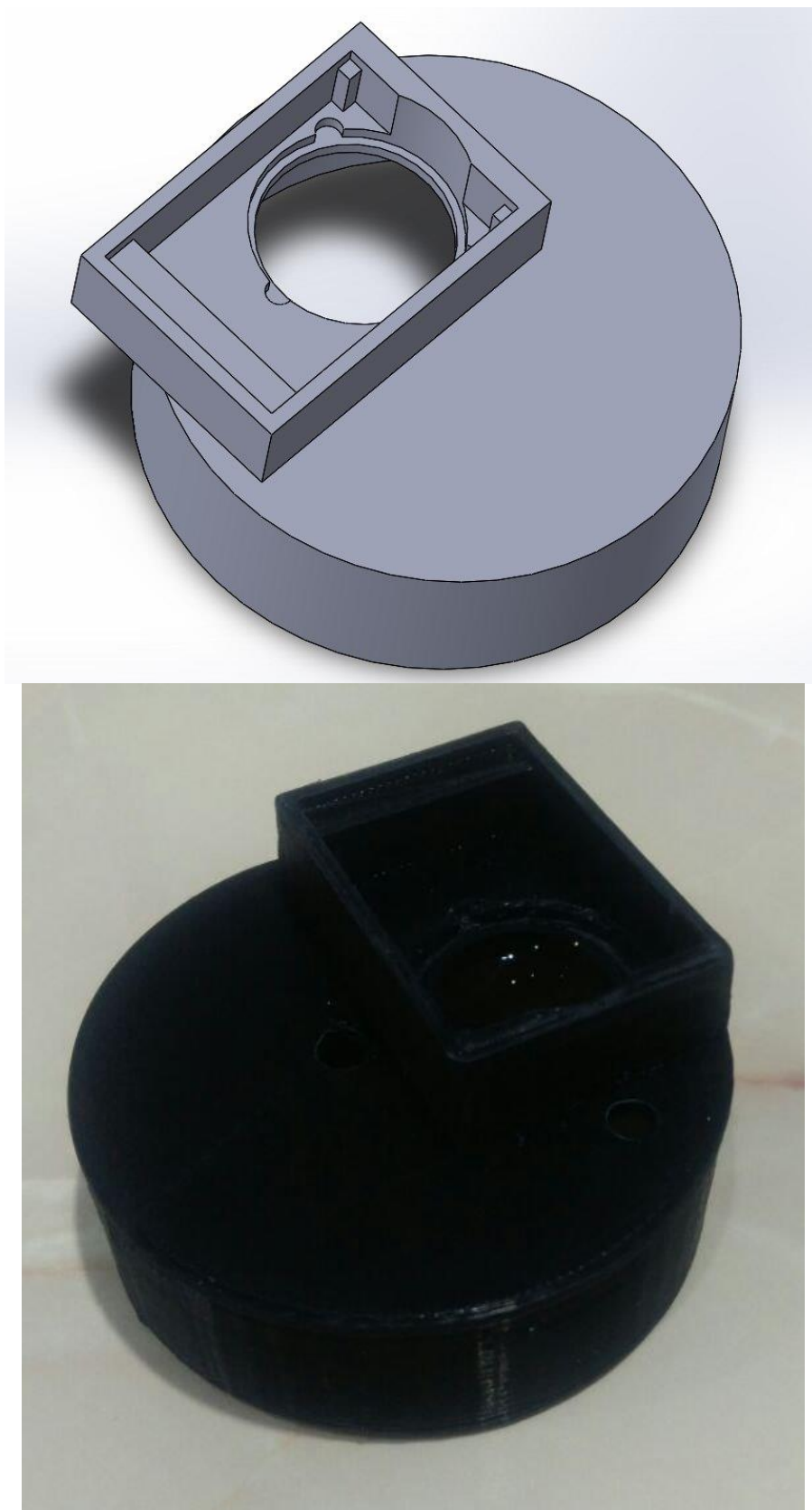


Рисунок 3.6 Кріплення камери з лінзою

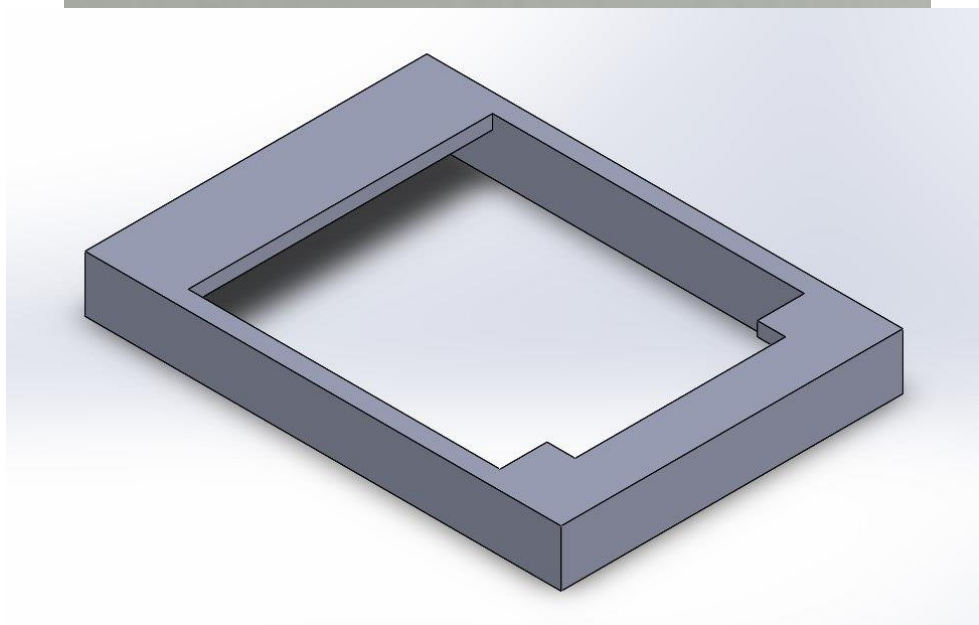
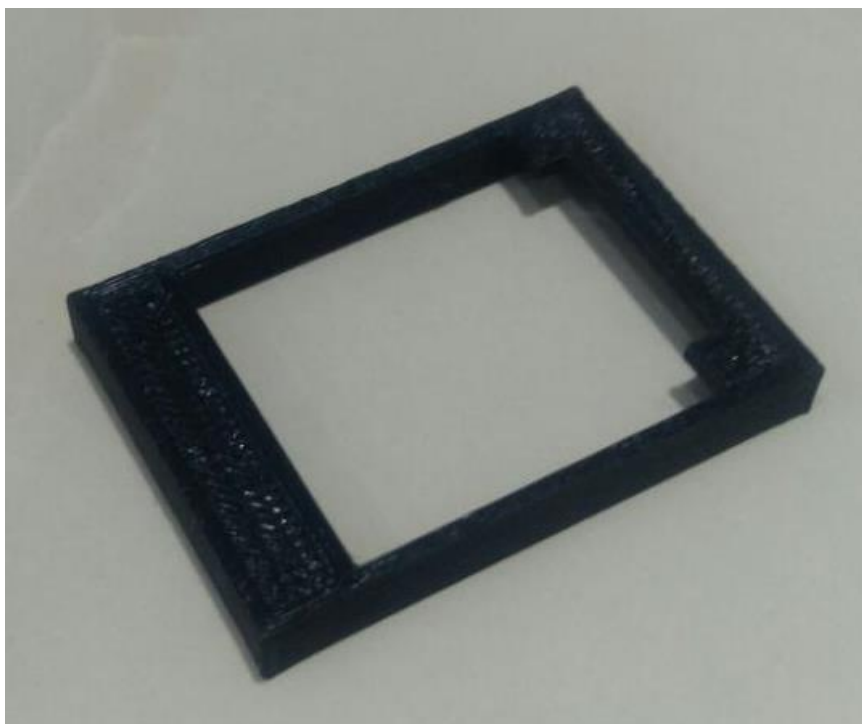


Рисунок 3.7 Фіксатор камери

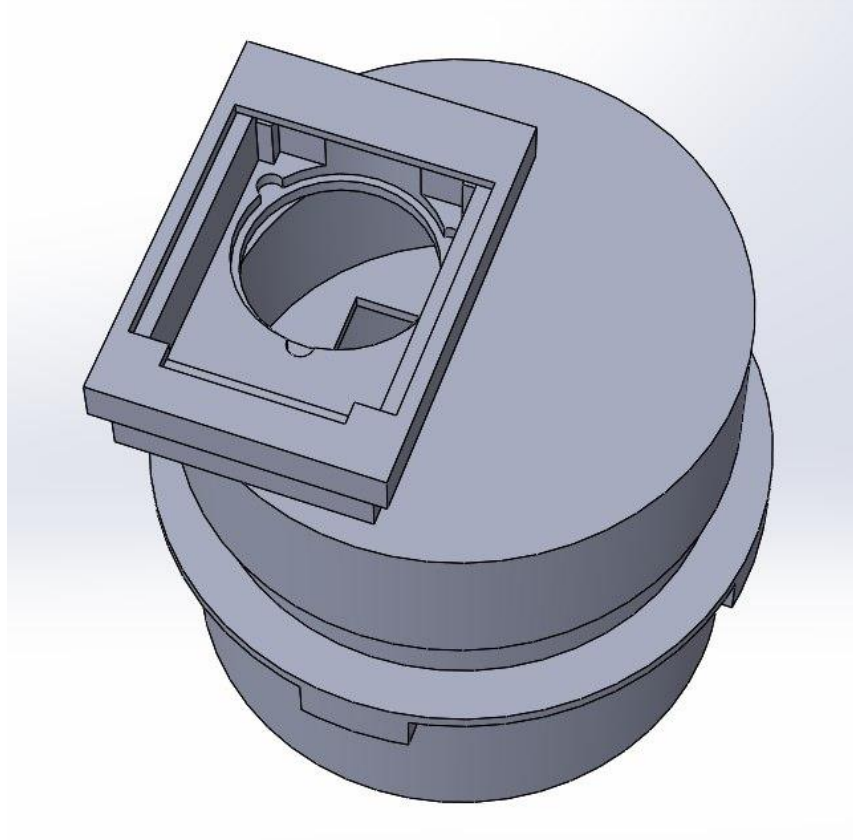


Рисунок 3.8 Зібраний комплект

РОЗДІЛ 4. РОБОТА СИТСЕМИ

4.1 Програмування ESP32-CAM

Для програмування плати була використана середа Arduino IDE 1.8.9.

Сам процес програмування відбувався за допомогою спеціального перехідника-програматора USB 2.0 в TTL UART та парочки дрітв-перемичек мама-мама.

Нижче представлений код скрипта:

```
#include "esp_http_client.h"
#include "esp_camera.h"
#include <WiFi.h>
#include "Arduino.h"
.

const char* ssid = "MVDK-WiFi";
const char* password = "750512matador1983";
int capture_interval = 5000; // Microseconds between captures
const char *post_url = "http://192.168.1.103/yourscript.php"; // Location
where images are POSTED

bool internet_connected = false;
long current_millis;
long last_capture_millis = 0;

// CAMERA_MODEL_AI_THINKER
#define PWDN_GPIO_NUM 32
```

```
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27
#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34

#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
#define Y3_GPIO_NUM      18
#define Y2_GPIO_NUM       5
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    23
#define PCLK_GPIO_NUM    22
```

```
void setup()
{

  pinMode (4, OUTPUT);

  Serial.begin(115200);

  if (init_wifi()) { // Connected to WiFi
    internet_connected = true;
    Serial.println("Internet connected");
  }
```

```

camera_config_t config;
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_0;
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
//init with high specs to pre-allocate larger buffers
if (psramFound()) {
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {

```

```

    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
}

}

bool init_wifi()
{
    int connAttempts = 0;
    Serial.println("\r\nConnecting to: " + String(ssid));
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
        if (connAttempts > 10) return false;
        connAttempts++;
    }
    return true;
}

esp_err_t _http_event_handler(esp_http_client_event_t *evt)

```



```

{
    switch (evt->event_id) {
        case HTTP_EVENT_ERROR:
            Serial.println("HTTP_EVENT_ERROR");
            break;
        case HTTP_EVENT_ON_CONNECTED:
            Serial.println("HTTP_EVENT_ON_CONNECTED");
            break;
        case HTTP_EVENT_HEADER_SENT:
            Serial.println("HTTP_EVENT_HEADER_SENT");
            break;
        case HTTP_EVENT_ON_HEADER:
            Serial.println();
            Serial.printf("HTTP_EVENT_ON_HEADER, key=%s, value=%s",
evt->header_key, evt->header_value);
            break;
        case HTTP_EVENT_ON_DATA:
            Serial.println();
            Serial.printf("HTTP_EVENT_ON_DATA, len=%d", evt->data_len);
            if (!esp_http_client_is_chunked_response(evt->client)) {
                // Write out data
                // printf("%.s", evt->data_len, (char*)evt->data);
            }
            break;
        case HTTP_EVENT_ON_FINISH:
            Serial.println("");
            Serial.println("HTTP_EVENT_ON_FINISH");
            break;
    }
}

```

```

case HTTP_EVENT_DISCONNECTED:
    Serial.println("HTTP_EVENT_DISCONNECTED");
    break;
}
return ESP_OK;
}

```

```

static esp_err_t take_send_photo()
{
    digitalWrite (4, HIGH); // ВКЛЮЧИТЬ СВЕТОДИОД
    delay (1000); // ПОДОЖДАТЬ 1 с

```

```

    Serial.println("Taking picture...");
    camera_fb_t * fb = NULL;
    esp_err_t res = ESP_OK;

```

```

    fb = esp_camera_fb_get();
    if (!fb) {
        Serial.println("Camera capture failed");
        return ESP_FAIL;
    }

```

```

    esp_http_client_handle_t http_client;

```

```

    esp_http_client_config_t config_client = {0};
    config_client.url = post_url;
    config_client.event_handler = _http_event_handler;
    config_client.method = HTTP_METHOD_POST;

```

```

http_client = esp_http_client_init(&config_client);

esp_http_client_set_post_field(http_client, (const char *)fb->buf, fb-
>len);

esp_http_client_set_header(http_client, "Content-Type", "image/jpg");

esp_err_t err = esp_http_client_perform(http_client);
if (err == ESP_OK) {
    Serial.print("esp_http_client_get_status_code: ");
    Serial.println(esp_http_client_get_status_code(http_client));
}

digitalWrite (4, LOW); // ВЫКЛЮЧИТЬ СВЕТОДИОД

esp_http_client_cleanup(http_client);

esp_camera_fb_return(fb);

ESP.deepSleep(20e6);
}

void loop()
{

```

```

// TODO check Wifi and reconnect if needed

current_millis = millis();
if (current_millis - last_capture_millis > capture_interval) { // Take
another picture
    last_capture_millis = millis();
    take_send_photo();
}
}

```

Як видно з коду виконується під'єднання до домашньої Wi-Fi мережі та виконується передача зробленого зображення по мережі на http сервер з заданою адресою. При цьому під час виконання фотографії спрацьовує спалах – окремо підключений світло діод. Після того, як картинка передана на сервер плата переходить в режим глибокого сну і заощаджує енергію.

4.2 Налаштування Raspberry

Спочатку необхідно підняти сервер, для того аби приймати файли від інших камер. Сервер буде працювати на двигуні Apache, оскільки це найрозповсюдженіший і найпотужніший двигун, який може працювати і налаштовуватись на raspberі.

“Apache – популярний додаток для веб-сервера, який можна встановити на Raspberry Pi, щоб воно могло обслуговувати веб-сторінки.

Apache самостійно може обслуговувати HTML-файли через HTTP, а за допомогою додаткових модулів можна обслуговувати динамічні веб-сторінки за допомогою мов скриптів, таких як PHP.

Встановити Apache

Спочатку оновіть доступні пакети, ввівши в Термінал таку команду:

```
sudo apt update
```

Потім встановіть пакет apache2 за допомогою цієї команди:

```
sudo apt install apache2 -y
```

Перевірте веб-сервер

За замовчуванням Apache розміщує тестовий HTML-файл у веб-папці. Ця веб-сторінка за замовчуванням подається під час переходу до `http: // localhost /` на самому Pi або `http://192.168.1.10` (незалежно від IP-адреси Pi) з іншого комп'ютера в мережі. Щоб знайти IP-адресу Pi, введіть ім'я хоста -I в командному рядку (або прочитайте докладніше про пошук вашої IP-адреси).

Перейдіть до веб-сторінки за замовчуванням або на Pi, або на іншому комп'ютері в мережі, і ви побачите таке:

Повідомлення про успіх Apache

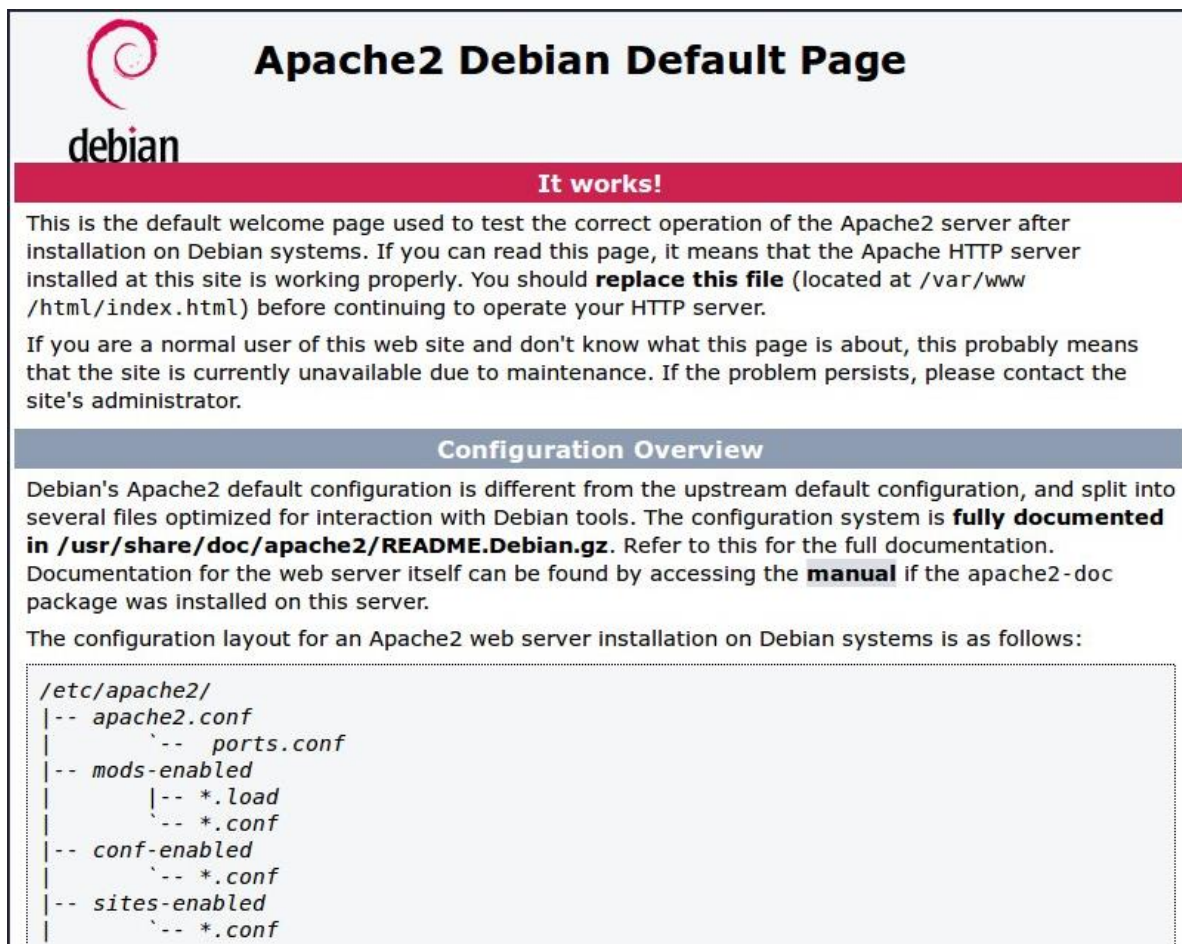


Рисунок 4.1 Встановлення Apache

Це означає, що у вас працює Apache!

Змінна веб-сторінка за замовчуванням

Ця веб-сторінка за замовчуванням - це лише файл HTML у файловій системі. Він розміщений за адресою `/var/www/html/index.html`.

Перейдіть до цього каталогу у вікні терміналу і подивіться, що знаходиться всередині:

```
cd / var / www / html
```

```
ls -al
```

Це покаже вам:

```
total 12
drwxr-xr-x  2 root root 4096 Jan  8 01:29 .
drwxr-xr-x 12 root root 4096 Jan  8 01:28 ..
-rw-r--r--  1 root root 177 Jan  8 01:29 index.html
```

Це показує, що за замовчуванням є один файл у / var / www / html / під назвою index.html . Для редагування файлу потрібно змінити його право власності на власне ім'я користувача. Змініть власника файлу (тут передбачається користувач pi за замовчуванням), використовуючи

```
sudo chown pi: index.html.
```

Тепер ви можете спробувати відредагувати цей файл, а потім оновити веб-переглядач, щоб побачити зміни веб-сторінки.

Ваш власний веб-сайт

Якщо ви знаєте HTML, ви можете помістити свої власні HTML-файли та інші активи в цей каталог та подати їх як веб-сайт у вашій локальній мережі.

Додатково - встановіть PHP

Щоб дозволити вашому серверу Apache обробляти файли PHP, вам потрібно буде встановити останню версію PHP та модуль PHP для Apache. Введіть таку команду, щоб встановити їх:

```
sudo apt install php libapache2-mod-php -y
```

Тепер видаліть файл index.html:

```
sudo rm index.html
```

і створити файл index.php:

```
sudo leafpad index.php” [7].
```

Тепер, маючи працюючий сервер необхідно його налаштувати. Для цього в папці /var/www/html створюємо папку timelapse та файл yoursript.php

В цьому файлі прописується наступне:

```
<?php
$received = file_get_contents('php://input');
$fileToWrite = "timelapse/upload - ".time().".jpg";
file_put_contents($fileToWrite, $received);
?>
```

Цей код дозволяє на сервері запустити прийом зображень, що надходять від «лічильників».

Тепер, коли є працюючий сервер і зображення зберігаються у папці то на пристрої можна зробити обробку фотографії.

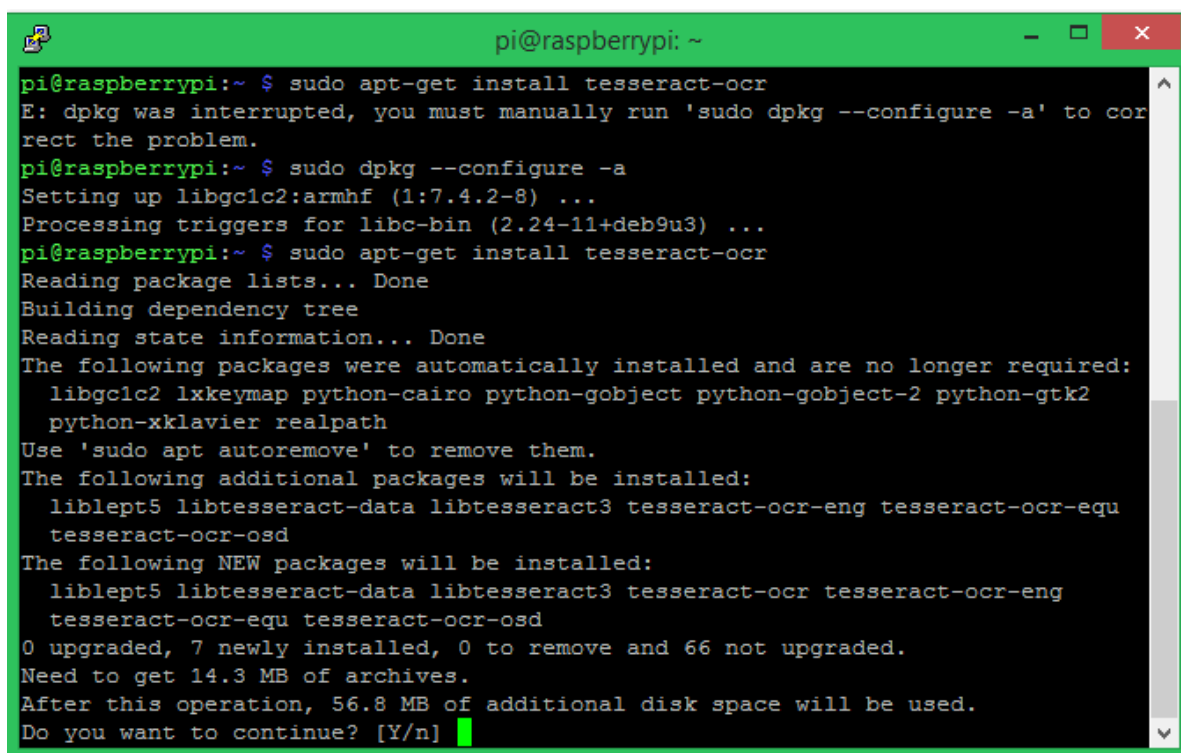
Щоб виконати оптичне розпізнавання символів на Raspberry Pi, нам доведеться встановити двигун OCR Tesseract на Pi. Для цього нам слід спочатку налаштувати пакет Debian (dpkg), який допоможе нам встановити OCR Tesseract. Використовуйте команду нижче у вікні терміналу, щоб налаштувати Debian Package.

```
sudo dpkg - -configure -a
```


Тоді ми можемо продовжити встановлення OCR Tesseract (оптичне розпізнавання символів) за допомогою опції apt-get. Команда для того ж наведена нижче.

```
sudo apt-get install tesseract-ocr
```

Термінальне вікно буде виглядати приблизно так, як нижче, для завершення установки знадобиться приблизно 5-10 хвилин.

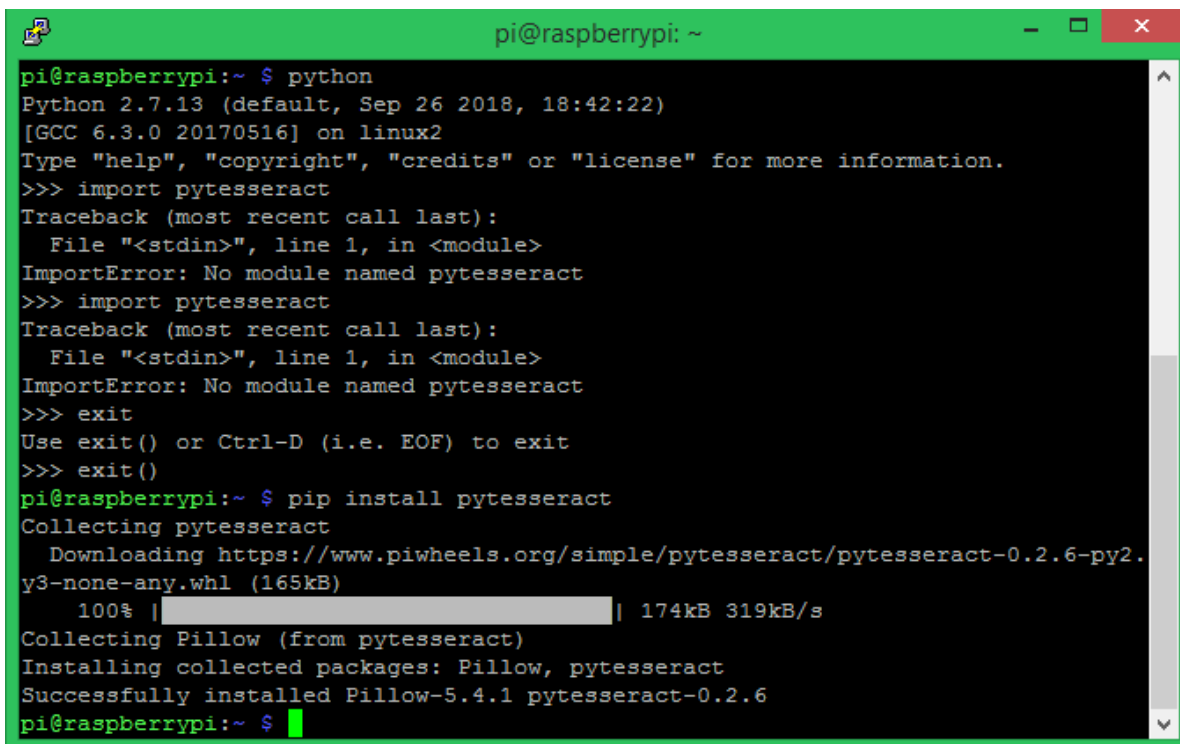


```
pi@raspberrypi:~ $ sudo apt-get install tesseract-ocr
E: dpkg was interrupted, you must manually run 'sudo dpkg --configure -a' to correct the problem.
pi@raspberrypi:~ $ sudo dpkg --configure -a
Setting up libc2:armhf (1:7.4.2-8) ...
Processing triggers for libc-bin (2.24-11+deb9u3) ...
pi@raspberrypi:~ $ sudo apt-get install tesseract-ocr
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libc2 lxkeymap python-cairo python-gobject python-gobject-2 python-gtk2
  python-xklavier realpath
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libtesseract3 tesseract-ocr-eng tesseract-ocr-equ tesseract-ocr-osd
The following NEW packages will be installed:
  libtesseract3 tesseract-ocr tesseract-ocr-eng tesseract-ocr-equ
0 upgraded, 7 newly installed, 0 to remove and 66 not upgraded.
Need to get 14.3 MB of archives.
After this operation, 56.8 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Рисунок 4.2 Оновлення пакетів

Тепер, коли у нас встановлено OCR Tesseract, нам доведеться встановити пакет PyTesseract за допомогою пакета встановлення pip. Pytesseract - це питонова обгортка навколо двигуна OCR tesseract, яка допомагає нам використовувати tesseract з python. Виконайте наведену нижче команду, щоб встановити pytesseract на python.

Pip install pytesseract



```

pi@raspberrypi:~ $ python
Python 2.7.13 (default, Sep 26 2018, 18:42:22)
[GCC 6.3.0 20170516] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import pytesseract
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named pytesseract
>>> import pytesseract
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named pytesseract
>>> exit
Use exit() or Ctrl-D (i.e. EOF) to exit
>>> exit()
pi@raspberrypi:~ $ pip install pytesseract
Collecting pytesseract
  Downloading https://www.piwheels.org/simple/pytesseract/pytesseract-0.2.6-py2.
y3-none-any.whl (165kB)
    100% |#####| 174kB 319kB/s
Collecting Pillow (from pytesseract)
Installing collected packages: Pillow, pytesseract
Successfully installed Pillow-5.4.1 pytesseract-0.2.6
pi@raspberrypi:~ $

```

Рисунок 4.3 Встановлення Тесеракт

Tesseract 4.0 для Windows / Ubuntu

Проект розпізнавання персонажів Tesseract Optical був спочатку розпочатий Hewlett Packard в 1980 році, а потім був прийнятий Google, який підтримує проект до сьогодні. Протягом багатьох років Tesseract розвивався, але все ще працює добре лише в контрольованих умовах. Якщо зображення має занадто багато фонового шуму або поза фокусом, тесеракт, здається, не працює добре.

Щоб подолати це, остання версія tesseract, Tesseract 4.0 використовує модель Deep Learning для розпізнавання символів і навіть почерку. Tesseract 4.0 використовує довгострокову пам'ять (LSTM) та

періодичну нейронну мережу (RNN), щоб підвищити точність роботи двигуна OCR. На жаль, через цей час цей підручник Tesseract 4.0 доступний лише для Windows та Ubuntu, але все ще знаходиться в бета-стадії для Raspberry Pi. Тому ми вирішили спробувати Tesseract 4.0 на windows та Tesseract 3.04 на Raspberry Pi.

Оскільки ми вже встановили пакети Tesseract OCR і Pytesseract в нашому PI. Ми можемо швидко написати невелику програму, щоб перевірити, як працює розпізнавання символів з тестовим зображенням. Тестове зображення, яке я використав, програму та результат можна знайти на зображенні нижче.

```
import pytesseract
from PIL import Image
import cv2

im = Image.open('5.jpg')
im.transpose(Image.ROTATE_180).save('6.jpg')

img = cv2.imread('6.jpg',cv2.IMREAD_COLOR) #Open the image from
which charectors has to be recognized
#img = cv2.resize(img, (620,480) ) #resize the image if required

gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY) #convert to grey to
reduce detials
gray = cv2.bilateralFilter(gray, 11, 17, 17) #Blur to reduce noise
```

```

original = pytesseract.image_to_string(gray, config='-l eng --oem 3 --psm 10')
#test = (pytesseract.image_to_data(gray, lang=None, config="", nice=0) ) #get
confidence level if required
#print(pytesseract.image_to_boxes(gray))
my_file = open(original+".txt", "w")
my_file.close()
print (original)

```

- Цей код виконує наступне:
- Перевертає зображення
- Обробляє по кольорам зображення
- Зменшує розмір для швидшої обробки
- Перетворює зображення в чорно-білий варіант
- Ефект Блур для зменшення шумів
- Запуск тезеракта з певними підібраними параметрами
- Створення і запис файлу з розпізнаними символами

Для автоматизації виконання цього скрипту використаємо crontab:

```
crontab -e
```

```
00,30 * * * * /home/var/www/html/timelaps/OCR.py
```

ВИСНОВКИ

В ході магістерської роботи було проведено фундаментальний аналіз усіх можливих варіантів створення системи автоматизації обліку енергоносіїв. Була обрана оптимальна модель для розробки. Практичним і теоретичним шляхом підбиралось обладнання для проекту. Було випробувано декілька різновидів обладнання різних виробників. Проаналізовано ринок збуту, доведена доцільність інноваційної розробки. Обрано найбільш дешевий та найбільш оптимальний з фінансової точки зору варіант по обладнанню. Було розроблену універсальну систему кріплення приладу на лічильники енергоносіїв та розроблена одна повністю працююча модель.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Системи ЛУЗОД/АСКОЕ [Електронний ресурс] URL: https://www.kyivenergy.top/ee-company/sistemi_luzodaskoe.html
2. Сайт компанії Вольт Енерго [Електронний ресурс] URL: <https://shop.voltenergo.com.ua/counter/po-energocentr>.
3. Программный комплекс АСКОЕ "Энергоцентр - 5" до 5 счетчиков вкл. [Електронний ресурс] URL: <https://shop.voltenergo.com.ua/counter/po-energocentr/pk-askoe-energotsentr-do-5-schetchikov>
4. Що таке Arduino? І чому це ідеальний старт для майбутнього розробника автоматизованих систем? [Електронний ресурс] URL: <https://keddr.com/2017/09/shho-take-arduino-i-chomu-tse-idealniy-start-dlya-maybutnogo-rozrobnika-avtomatizovanih-sistem/>
5. Інтернет-магазин 1wire [Електронний ресурс] URL: <https://1wire.com.ua/esp32-cam.html>
6. Raspberry Pi 3 Model B [Електронний ресурс] URL: <http://arduino.ua/prod1449-raspberry-pi-3-b>
7. Setting up an Apache Web Server on a Raspberry Pi [Електронний ресурс] URL: <https://www.raspberrypi.org/documentation/remote-access/web-server/apache.md>